# US Government Protection Profile

# for

# *Database Management Systems in Basic Robustness Environments*

**National Security Agency**
**Information Assurance Directorate**

**Version 0.24**

**15 December 2003**

# Foreword

1 This document, "US Government Protection Profile for Database Management Systems in Environments Requiring Basic Robustness" (DBMS-BASIC PP), will be issued by the National Security Agency (in final form) as part of its program to promulgate standards for information systems security. The base set of requirements used in this protection profile is taken from the "Common Criteria for Information Technology Security Evaluation, Version 2.1."

2 Comments on this document should be directed to: ppcomments@iatf.net. The comments should include the title of the document, the page, the section number, and paragraph number, detailed comment and recommendations.

# Table of Contents

# List of Tables and Figures

# 1 Introduction

3 This section contains overview information necessary to allow a Protection Profile (PP) to be registered through a Protection Profile Registry. The PP identification provides the labeling and descriptive information necessary to identify, catalogue, register, and cross-reference a PP. The PP overview summarizes the profile in narrative form and provides sufficient information for a potential user to determine whether the PP is of interest. The overview can also be used as a stand-alone abstract for PP catalogues and registers. The "Conventions" section provides the notation, formatting, and conventions used in this protection profile. The "Glossary of Terms" section gives a basic definition of terms, which are specific to this PP. The "Document Organization" section briefly explains how this document is organized.

## 1.1 Identification

4 Title: U.S. Government Protection Profile for Database Management Systems in Environments Requiring Basic Robustness, Version 0.24, 15 December 2003.

5 Registration: <to be provided upon registration>

6 Keywords: database management system, DBMS, COTS, commercial security, basic robustness, access control, discretionary access control, DAC, CC EAL2 augmented

## 1.2 Overview

7 The "*U.S. Government Protection Profile for Database Management Systems in Environments Requiring Basic Robustness"* specifies security requirements for a commercial-off-the-shelf (COTS) database system that includes but is not limited to a DBMS server and may be evaluated as a software only application layered on an underlying system (i.e., operating system, hardware, network services and/or custom software) and is usually embedded as a component of a larger system within an operational environment. This profile establishes the requirements necessary to achieve the security objectives of the Target of Evaluation (TOE) and its environment.

8 Conformant products provide access control based on user identity (e.g., Discretionary Access Control (DAC)) and generation of audit records for security relevant events. A conformant product or its IT environment provides the following functionality: identification and authentication, security administration and audit record storage, and audit review. A conformant product, in conjunction with its IT environment that satisfies all the requirements in this protection profile, provides necessary security services, mechanisms, and assurances to process administrative, private, and sensitive/proprietary information. The intended environment for conformant products has a relatively low threat for the sensitivity of the data processed. Authorized users, including authorized administrators, of the TOE generally are

trusted not to attempt to circumvent access controls implemented by the TOE to gain access to data for which they are not authorized.

### 1.2.1 TOE Environment Defining Factors

9 In trying to specify the environments in which TOEs with various levels of robustness are appropriate, it is useful to first discuss the two defining factors that characterize that environment: **value of the TOE and the data it contains** and **authorization of the entities with access to the TOE** to those resources.

10 In general terms, the environment for a TOE can be characterized by the authorization (or lack of authorization) the least trustworthy entity has with respect to the highest value of TOE resources (i.e. the TOE itself and all of the data processed by the TOE).

11 Note that there are an infinite number of combinations of entity authorization and value of resources; this conceptually "makes sense" because there are an infinite number of potential environments, depending on how the resources are valued by the organization, and the variety of authorizations the organization defines for the associated entities. In the next section 1.2.2, these two environmental factors will be related to the robustness required for selection of an appropriate TOE.

#### 1.2.1.1 VALUE OF TOE AND THE DATA IT CONTAINS

12 Value of the resources associated with the TOE includes the data being processed or used by the TOE, as well as the TOE itself (for example, a real-time control processor). "Value" is assigned by the using organization. For example, in the DoD low-value data might be equivalent to data marked "For Official Use Only", while high-value data may be those classified Top Secret. In a commercial enterprise, low-value data might be the internal organizational structure as captured in the corporate on-line phone book, while high-value data might be corporate research results for the next generation product.

13 Value of the TOE and its data can also be thought of in terms of the negative publicity that would occur if the TOE or the data it hosts were to be compromised. An online auction site that has a database of items for sale must protect the integrity of that database, though much of the information is for public consumption. If customers of that site lose confidence that the auction will function as intended, then the operators of that site would lose credibility, and their business of conducting online auctions would be in dire jeopardy.

#### 1.2.1.2 AUTHORIZATION OF ENTITIES WITH ACCESS TO THE TOE

14 Authorization that entities (users, administrators, other IT systems) have with respect to the TOE (and thus the resources of that TOE, including the TOE itself) is an abstract concept reflecting a combination of the trustworthiness of an entity and the access and privileges granted to that entity with respect to the resources of the TOE. For instance, entities that have total authorization to all data on the TOE are at one end of this spectrum; these entities may have privileges that allow them to read, write, and modify anything on the TOE, including all data used by the TOE security functions. Entities at the other end of the spectrum are those that are authorized to few or no TOE resources. For example, in the case

of a DBMS, an entity may not be allowed to log on to the TOE at all (that is, they are not valid users listed in the DBMS's user database).

15 It is important to note that authorization **does not** refer to the **access** that the entities actually have to the TOE or its data. For example, suppose the owner of the system determines that no one other than employees are authorized to certain data on a TOE, yet they connect the TOE to the Internet. There are millions of entities that are not **authorized** to the data (because they are not employees), but they actually have connectivity to the TOE through the Internet and thus can attempt to access the TOE and its associated resources.

16 Continuing with the DBMS example, a DBMS that is compliant with this protection profile might host public objects, containing data that anyone may read. No **authorization** is required to read public objects, only **access**. If the DBMS is on the Internet, anyone with connectivity to the DBMS may observe the data contained in the public objects. If the DBMS is part of a system for an online auction, the public objects might contain the listing of items for sale, the descriptions of the items, the current bid price, and so on. There are other non-public objects that require authorization. The non-public objects might contain sensitive data such as credit card numbers for the auction customers and reserve prices of items, which are to be known only by the seller.

17 Entities are characterized according to the value of resources to which they are authorized; the extent of their authorization is implicitly a measure of how trustworthy the entity is with respect to compromise of the data (that is, compromise of any of the applicable security policies; e.g., confidentiality, integrity, availability). In other words, in this model the greater the extent of an entity's authorization, the more trustworthy (with respect to applicable policies) that entity is.

### 1.2.2 Selection of Appropriate Robustness Levels

18 Robustness is a characteristic of a TOE defining how well it can protect itself and its resources; a more robust TOE is better able to protect itself. This section relates the defining factors of IT environments, authorization, and value of resources to the selection of appropriate robustness levels.

19 When assessing any environment with respect to Information Assurance the critical point to consider is the likelihood of an attempted security policy compromise, which was characterized in the previous section in terms of entity authorization and resource value. As previously mentioned, robustness is a characteristic of a TOE that reflects the extent to which a TOE can protect itself and its resources. It follows that as the likelihood of an attempted resource compromise increases, the robustness of an appropriate TOE should also increase.

20 It is critical to note that several combinations of the environmental factors will result in environments in which the likelihood of an attempted security policy compromise is similar. Consider the following two cases:

21 The first case is a TOE that processes only low-value data. Although the organization has stated that only its employees are authorized to log on to the system and access the data, the system is connected to the Internet to allow authorized employees to access the system from home. In this case, the least trusted entities would be unauthorized entities (e.g. non-employees) exposed to the TOE because of the Internet connectivity. However, since only low-value data are being processed, the likelihood that unauthorized entities would find it worth their while to attempt to compromise the data on the system is low and selection of a basic robustness TOE would be appropriate.

22 The second case is a TOE that processes high-value (e.g., classified) information. The organization requires that the TOE be isolated from other IT systems processing lower-value information, and that every user with physical and logical access to the TOE undergo an investigation so that they are authorized to the highest-value data on the TOE. Because of the extensive checks done during this investigation and the physical protection afforded the TOE, the organization is assured that only highly-trusted users are authorized to use the TOE. In this case, even though high-value information is being processed, it is unlikely that a compromise of that data will be attempted because of the authorization and trustworthiness of the users and once again, selection of a basic robustness TOE would be appropriate.

23 As a third case, consider the DBMS that is part of the online auction system. Much of the data it hosts is public data about the items for sale. However, the value of that data is very high. It is critical to the success of the online auction business. In military parlance, one could say the public data is “mission critical.” Basic robustness would not be an appropriate choice for a DBMS housing such commercial high-value data.

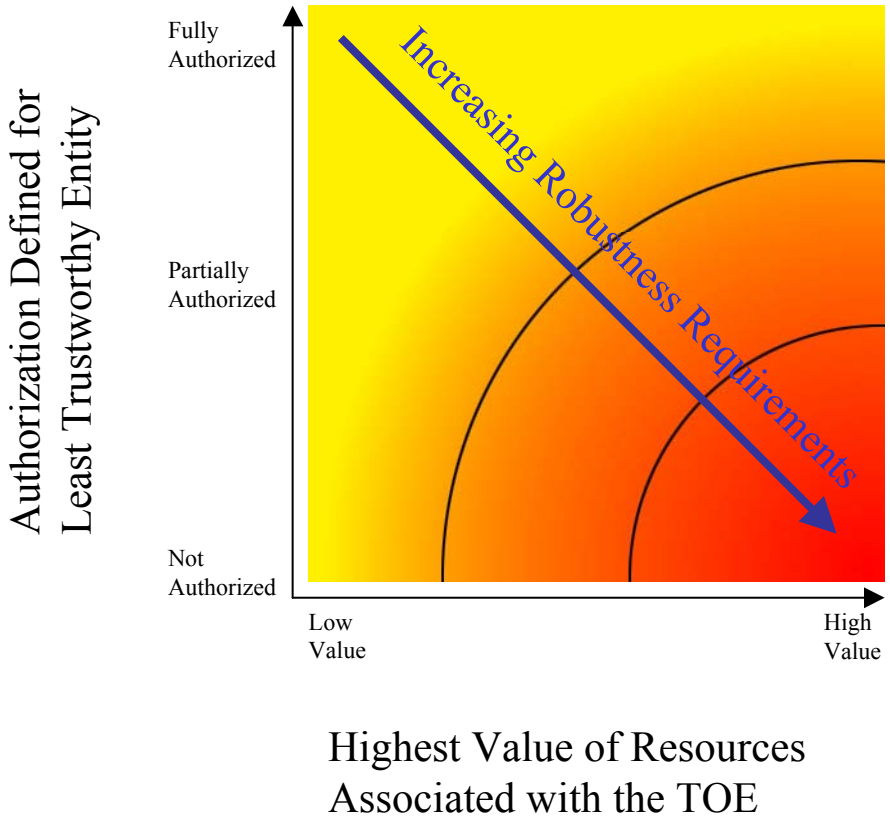
24 The preceding examples demonstrated that it is possible for radically different combinations of entity authorization and resource values to result in a similar likelihood of an attempted compromise. As mentioned earlier, the robustness of a system is an indication of the protection being provided to counter compromise attempts. Therefore, a basic robustness system should be sufficient to counter compromise attempts where the likelihood of an attempted compromise is low. Figure 1 depicts the “universe” of environments characterized by the two factors discussed in the previous section: on one axis is the authorization defined for the least trustworthy entity, and on the other axis is the highest value of resources associated with the TOE.

25 As depicted in figure 1, the robustness of the TOEs required in each environment steadily increases as one goes from the upper left of the graph to the lower right; this corresponds to the need to counter increasingly likely attack attempts by the least trustworthy entities in the environment. Note that the shading of the graph is intended to reflect the notion that different environments engender similar levels of “likelihood of attempted compromise”, signified by a similar color. Further, the delineations between such environments are not stark, but rather are finely grained and gradual.

26 While it would be possible to create many different "levels of robustness" at small intervals along the “Increasing Robustness Requirements” line to counter the increasing likelihood of attempted compromise due to those attacks, it would not be practical or particularly useful. Instead, in order to implement the robustness strategy where there are only three robustness

levels: Basic, Medium, and High, the graph is divided into three sections, with each section corresponding to set of environments where the likelihood of attempted compromise is roughly similar. This is graphically depicted in the Figure 1.

**Figure 1 - Universe of Environments**

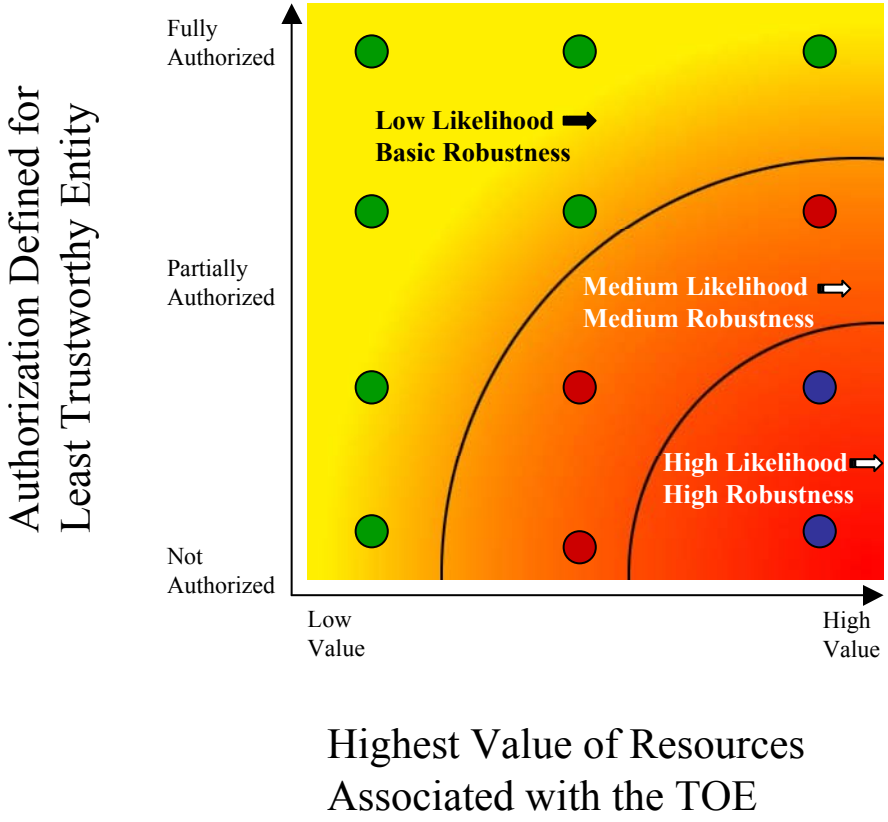



27 In this second representation of environments and the robustness plane below, Figure 2, the "dots" represent given instantiations of environments; arched lines define environments with a similar likelihood of attempted compromise. Correspondingly, a TOE with a given robustness should provide sufficient protection for environments characterized within these arched lines. In choosing the appropriateness of a given robustness level TOE PP for an environment, then, the user must first consider the lowest authorization for an entity as well as the highest value of the resources in that environment. This should result in a "point" in the graph above, corresponding to the likelihood that that entity will attempt to compromise

the most valuable resource in the environment. The appropriate robustness level for the specified TOE to counter this likelihood can then be chosen.

28 The difficult part of this activity is differentiating the authorization of various entities, as well as determining the relative values of resources (e.g., what constitutes "low value" data vs. "medium value" data). Because every organization will be different, a rigorous definition is not possible. In Section 3 of this PP, the targeted threat level for a basic robustness TOE is characterized. This information is provided to help organizations using this PP insure that the functional requirements specified by this basic robustness PP are appropriate for their intended application of a compliant TOE.

**Figure 2 - Likelihood of Attempted Compromise**




## 1.3 Conventions

29 The notation, formatting, and conventions used in this protection profile (PP) are consistent with version 2.1 of the Common Criteria for Information Technology Security Evaluation. Font style and clarifying information conventions were developed to aid the reader.

30 The CC permits four functional component operations: assignment, iteration, refinement, and selection to be performed on functional requirements. These operations are defined in Common Criteria, Part 2, paragraph 2.1.4 as:

- Assignment: allows the specification of an identified parameter;
- Refinement: allows the addition of details or the narrowing of requirements;
- Selection: allows the specification of one or more elements from a list; and
- Iteration: allows a component to be used more than once with varying operations.

31 *Assignments or selections* that are left to be specified by the developer in subsequent security target documentation, and are italicized and identified between brackets ("[ ]"). In addition, when an assignment or selection has been left to the discretion of the developer, the text "assignment:" or "selection:" is indicated within the brackets. Assignments or selection created by the PP author (for the developer to complete) are bold, italicized, and between brackets ("[ ]"). CC selections completed by the PP author are underlined and CC assignments completed by the PP author are bold.

32 *Refinements* are identified with "**Refinement:**" right after the short name. They permit the addition of extra detail when the component is used. The underlying notion of a refinement is that of narrowing. There are two types of narrowing possible: narrowing of implementation and narrowing of scope[1]. Additions to the CC text are specified in bold. Deletions of the CC text are identified in the "End Notes" with a bold number after the element ("**8**").

33 *Iterations* are identified with a number inside parentheses ("(#)"). These follow the short family name and allow components to be used more than once with varying operations.

34 *Explicit Requirements* are allowed to create requirements should the Common Criteria not offer suitable requirements to meet the PP needs. The naming convention for explicit requirements is the same as that used in the CC. To ensure these requirements are explicitly identified, the ending "**_EXP**" is appended to the newly created short name.

35 *Application Notes* are used to provide the reader with additional requirement understanding or to clarify the author's intent. These are italicized and usually appear following the element needing clarification.

36 These conventions are expressed by using combinations of bolded, italicized, and underlined text as specified in Table 1.

[1] US interpretation #0362: Scope of Permitted Refinements

**Table 1 - Functional Requirements Operation Conventions**

| Convention | Purpose | Operation |
|---|---|---|
| **Bold** | The purpose of bolded text is used to alert the reader that additional text has been added to the CC. This could be an assignment that was completed by the PP author or a refinement to the CC statement.<br>Examples:<br>FAU_SAR.1.1 The TSF shall provide **authorized administrators** with the capability to read **all audit information** from the audit records.<br>FTA_MCS.1.1 **Refinement:** The TSF shall restrict the maximum number of concurrent **interactive** sessions that belong to the same user. | (Completed) Assignment<br>or<br>Refinement |
| *Italics* | The purpose of italicized text is to inform the reader of an assignment or selection operation to be completed by the developer or ST author. It has been left as it appears in the CC requirement statement.<br>Examples:<br>FTA_SSL.1.1The TSF shall lock an interactive session after *[assignment: a time interval of user inactivity]* by:<br>a) Clearing or overwriting display devices, making the current contents unreadable.<br>b) Disabling any activity of the user's data access/display devices other than unlocking the session.<br>FDP_RIP.2.1 **Refinement:** The TSF shall ensure that any previous information content of a resource is made unavailable upon the *[selection: allocation of the resource to, deallocation of the resource from]* all objects **other than those associated with cryptographic keys and critical cryptographic security parameters as described in FCS_CKM.4.1 and FCS_CKM_EXP.2.5**. | Assignment (to be completed by developer or ST author)<br>or<br>Selection (to be completed by developer or ST author) |

| Convention | Purpose | Operation |
|---|---|---|
| Underline | The purpose of underlined text is to inform the reader that a choice was made from a list provided by the CC selection operation statement.<br>Example:<br>FAU_STG.1.2 The TSF shall be able to prevent modifications to the audit records. | Selection (completed by PP author) |
| ***Bold & Italics*** | The purpose of bolded and italicized text is to inform the reader that the author has added new text to the requirement and that an additional vendor action needs to be taken.<br>Example:<br>FIA_UAU.1.1 **Refinement:** The TSF shall allow **read access to *[assignment: list of public objects]*** on behalf of the user to be performed before the user is authenticated.<br>FCS_CKM.2.1 – The TSF shall distribute cryptographic keys in accordance with a specified cryptographic key distribution method ***[selection: Manual (Physical) Method, Automated (Electronic Method), Manual Method and Automated Method]*** that meets the … | Assignment (added by the PP author for the developer or ST author to complete)<br>or<br>Selection (added by the PP author for the developer or ST author to complete)) |

| Convention | Purpose | Operation |
|---|---|---|
| Parentheses<br>(Iteration #) | The purpose of using parentheses and an iteration number is to inform the reader that the author has selected a new field of assignments or selections with the same requirement and that the requirement will be used multiple times. Iterations are performed at the component level. The component behavior name includes information specific to the iteration between parentheses.<br>Example:<br>**5.5.3.1 Management of TSF Data (for general TSF data) (FMT_MTD.1(1))**<br>FMT_MTD.1.1(1) The TSF shall restrict the ability to **create**, query, modify, delete, and clear the **security-relevant TSF data except for audit records, user security attributes, and authentication data** to **the authorized administrator**.<br>**5.5.3.2 Management of TSF Data (for audit records) (FMT_MTD.1(2))**<br>FMT_MTD.1.1(2) The TSF shall restrict the ability to query, delete, and clear the **audit records** to **authorized administrators**. | Iteration 1 (of component)<br>Iteration 2 (of component) |
| Explicit: (**_EXP**) | The purpose of using **Explicit:** before the family or component behavior name is to alert the reader and to explicitly identify a newly created component. To ensure these requirements are explicitly identified, the "**_EXP**" is appended to the newly created short name and the family or component name is bolded.<br>Example:<br>**5.5.7.1 EXPLICIT: INTERNAL TSF DATA CONSISTENCY (FPT_TRC_EXP.1)**<br>**FPT_TRC_EXP.1.1** The TSF shall ensure that TSF data is consistent between parts of the TOE by providing a mechanism to bring inconsistent TSF data into a consistent state in a timely manner. | Explicit Requirement |

| Convention | Purpose | Operation |
|---|---|---|
| Endnotes | The purpose of endnotes is to alert the reader that the author has deleted Common Criteria text. An endnote number is inserted at the end of the requirement, and the endnote is recorded on the last page of the section. The endnote statement first states that a deletion was performed and then provides the rationale. Following is the family behavior or requirement in its original and modified form. A strikethrough is used to identify deleted text and bold for added text. A text deletion rationale is provided. Examples:<br>Text as shown:<br>FPT_TST.1.2 **Refinement:** The TSF shall provide authorized **administrators** with the capability to verify the integrity of TSF data.**18**<br>Endnote statement:<br>**18** A deletion of CC text was performed in FPT_TST.1.2. Rationale: The word " users " was deleted to replace it with the role of "authorized administrator". Only authorized administrators should be given the capability to verify the integrity of the TSF data.<br>FPT_TST.1.2 **Refinement:** The TSF shall provide authorized ~~users~~ **administrators** with the capability to verify the integrity of TSF data. | Refinement |

# 1.4 Glossary of Terms

37 This profile uses the terms described in this section to aid in the application of the requirements.

**Table 2 - Glossary of Terms**

| | |
|---|---|
| Access | Interaction between an entity and an object that results in the flow or modification of data. |
| Access Control | Security service that controls the use of resources and the disclosure and modification of data. |
| Accountability | Property that allows activities in an IT system to be traced to the entity responsible for the activity. |
| Attack | An intentional act attempting to violate the security policy of an IT system. |
| Authentication | Security measure that verifies a claimed identity. |
| Authorization | Permission, granted by an entity authorized to do so, to perform functions and access data. |

| | |
|---|---|
| Authorized Administrator | An authorized person in contact with the Target of Evaluation who is responsible for maintaining its operational capability. |
| Availability | Timely, reliable access to IT resources. |
| Compromise | Violation of a security policy. |
| Confidentiality | A security policy pertaining to disclosure of data. |
| Conformant Product | A Target of Evaluation that satisfies all the functional security requirements in Section 5.1. The requirements in section 5.2 are satisfied by either the TOE or its IT environment. And the requirements in section 5.3 are satisfied by its IT environment. Furthermore, a conformant TOE satisfies all the TOE security assurance requirements in section 5.4 of this document. |
| Authorized administrator | A user who has been granted the authority to manage the TOE and whose actions may affect the TSP. Administrators may possess special privileges that provide capabilities to override portions of the TSP. |
| Database Management System (DBMS) | A suite of programs that typically manage large structured sets of persistent data, offering ad hoc query facilities to many users. They are widely used in business applications. |
| Discretionary Access Control (DAC) | A means of restricting access to objects based on the identity of subjects or groups to which they belong. These controls are discretionary in the sense that a subject with a certain access permission is capable of passing that permission (perhaps indirectly) on to any other subject. |
| Enclave | A collection of entities under the control of a single authority and having a homogeneous security policy. They may be logical, or may be based on physical location and proximity. |
| Entity | A subject, object, user, or another IT device, which interacts with TOE objects, data, or resources. |
| Integrity | A security policy pertaining to the corruption of data and TSF mechanisms. |

| | |
|---|---|
| Named object | An object that exhibits all of the following characteristics:<br>- The object may be used to transfer information between subjects of differing user identities within the TSF.<br>- Subjects in the TOE must be able to request a specific instance of the object.<br>- The name used to refer to a specific instance of the object must exist in a context that potentially allows subjects with different user identities to request the same instance of the object. |
| Operating environment | The total environment in which a TOE operates. It includes the physical facility and any physical, procedural, administrative and personnel controls. |
| Public object | An object for which the TSF unconditionally permits all entities "read" access. No authorization is required to read public objects. All that is necessary is access to the TOE, whether it be physical access or access via a network. |

| | |
|---|---|
| Robustness | A characterization of the strength of a security function, mechanism, service or solution, and the assurance (or confidence) that it is implemented and functioning correctly. DoD has three levels of robustness:<br><br>• **Basic:** Security services and mechanisms that equate to good commercial practices. Basic robustness equates to EAL-2 plus; ALC_FLR (Flaw Remediation), and AVA_MSU.1 (Misuse-Examination Guidance) as defined in CCIB-98-028, Part 3, Version 2.0<br><br>• **Medium:** Security services and mechanisms that provide for layering of additional safeguards above good commercial practices. Medium robustness equates to EAL-4 plus; AMA (Maintenance of Assurance); ALC_FLR (Flaw Remediation); ADV_IMP.2; ADV_INT.1; ATE_DPT.2; and AVA_VLA.3 (Moderately Resistant Vulnerability Analysis) as defined in CCIB-98-028, Part 3, Version 2.0. If cryptographic functions are included in the TOE, then the PP should be augmented with AVA_CCA_EXP.2 as documented in the Protection Profile Medium Robustness Consistency Guidance.<br><br>• **High:** Security services and mechanisms that provide the most stringent protection and rigorous security countermeasures. |
| Secure state | Condition in which all TOE security policies are enforced. |
| Sensitive information | Information that, as determined by a competent authority, must be protected because its unauthorized disclosure, alteration, loss, or destruction will at least cause perceivable damage to someone or something. |
| Threat | Capabilities, intentions and attack methods of adversaries, or any circumstance or event, with the potential to violate the TOE security policy. |
| Unauthorized user | A user who may obtain access only to system provided public objects if any exist. |
| Vulnerability | A weakness that can be exploited to violate the TOE security policy. |

## 1.5 Document Organization

38 *Section 1* provides the introductory material for the protection profile.

39 *Section 2* describes the Target of Evaluation in terms of its envisaged usage and connectivity.

40 *Section 3* defines the expected TOE security environment in terms of the threats to its security, the security assumptions made about its use, and the security policies that must be followed.

41 *Section 4* identifies the security objectives derived from these threats and policies.

42 *Section 5* identifies and defines the security functional requirements from the CC that must be met by the TOE and the IT environment in order for the functionality-based objectives to be met. This section also identifies the security assurance requirements for EAL2 augmented.

43 *Section 6* provides a rationale to explicitly demonstrate that the information technology security objectives satisfy the policies and threats. Arguments are provided for the coverage of each policy and threat. The section then explains how the set of requirements are complete relative to the objectives, and that each security objective is addressed by one or more component requirement. Arguments are provided for the coverage of each objective.

44 *Section 7* identifies background material used as reference to create this Protection Profile.

# 2 TOE Description

## 2.1 Product Type

45 The product type of the Target of Evaluation described in this PP is a database management system (DBMS) with the capability to limit TOE access to authorized users, enforce Discretionary Access Controls on objects under the control of the database management system based on user authorizations, and to provide user accountability via audit of users' actions.

46 A DBMS is a computerized repository that stores information and allows authorized users to retrieve and update that information. A DBMS may be a single-user system, in which only one user may access the DBMS at a given time, or a multi-user system, in which many users may access the DBMS simultaneously.

47 A DBMS supports two major types of users:

- Users who interact with the DBMS to observe and/or modify data objects for which they have authorization to access;
- Authorized administrators who implement and manage the various information-related policies of an organization (e.g., access, integrity, consistency, availability) on the databases that they manage and/or own.

48 A DBMS, in conjunction with the IT environment, stores, and controls access to, two types of data:

- The user data that the DBMS maintains and protects. User data may consist of the following:
  - The user data stored in or as database objects;
  - The definitions of user databases and database objects, commonly known as DBMS metadata;
  - User-developed queries, functions, or procedures that the DBMS maintains for users.
- The DBMS data (e.g., configuration parameters, user security attributes, transaction log, audit instructions and records) that the DBMS maintains and uses to operate the DBMS.

49 Most commercial DBMSs have the following major components:

- The DBMS server application that performs the following functions:

- Controlling users' accesses to user data and DBMS data;
- Interacting with, and possibly supplementing portions of, the underlying operating system to retrieve and present the data that are under the DBMS's management;
- Indexing data values to their physical locations for quick retrievals based on a value or range of values;
- Executing pre-written programs (i.e., utilities) to perform common tasks like database backup, recovery, loading, and copying;
- Supporting mechanisms that enable concurrent database access (e.g., locks);
- Assisting recovery of user data and DBMS data (e.g., transaction log); and
- Tracking operations performed by users.

- A database client application through which DBMS users interact with the DBMS server (e.g., direct queries, stored procedures);
- A data model with which the DBMS data structures and organization can be conceptualized (e.g., hierarchical, object-oriented, relational data models) and DBMS objects defined; and
- High-level language(s) or interfaces that allow authorized users to define database constructs; access and modify user or DBMS data; present user or DBMS data; and perform operations on those data.

50 A DBMS specification is the proper document in which to identify the detailed requirements for the DBMS manager/server functions listed above (and any additional DBMS functions). This PP identifies the requirements for the security functions that the DBMS performs in addition to, or as part of, those DBMS manager/server functions. This PP also identifies security requirement for the IT environment in which the DBMS operates.

## 2.2 TOE Definition

51 The TOE consists of at least one instance of the DBMS server application with its associated guidance documentation and the interfaces to the external IT entities with which the DBMS interacts. Other components that may or may not be included in the TOE are:

- Any database clients that allow users to interface with the DBMS server
- Any middleware required for the DB server and/or clients to run
- Host operating system(s) and underlying hardware that server(s) and client(s) require to run on

52 This PP does not dictate a specific architecture. The architecture of the TOE can be a distributed or a non-distributed. The TOE data may reside on a single host or be distributed

among several hosts. If the TOE is a distributed architecture, the TOE may depend on the IT environment to provide adequate protection, whether through physical or cryptographic means, to transmit user and DBMS data between the components comprising the TOE. The vendor will have to identify and describe the TOE architecture that they will evaluate.

53 The external IT entities with which the DBMS may interact—if they are outside the TOE—include the following:

- Client applications that allow users to interface with the DBMS server
- The host operating system (host OS) on which the TOE has been installed;
- The networking, printing, data-storage, and other devices and services with which the host OS may interact on behalf of the DBMS or the DBMS user; and
- The other IT products such as application servers, web servers, authentication servers, audit servers, and transaction processors with which the DBMS may interact to perform a DBMS function or a security function.

54 If the host OS is outside the TOE, the DBMS must specify the host OS on which it must reside to provide the desired degree of security feature integration. However, the goals of confidentiality, integrity and availability for the TOE must be met by the total package: the DBMS and the external IT entities with which it interacts. In all cases the TOE must be installed and administered in accordance with the TOE installation and administration instructions.

## 2.3 General TOE Security Functionality

55 A DBMS evaluated against this PP will provide the following security services either completely or in cooperation with the IT environment:

56 Security services that must be provided by the TOE:

- Discretionary Access Control (DAC) which controls access to objects based on the identity of the subjects or groups to which the subjects and objects belong, and which allows authorized users to specify how the objects that they control are protected.
- Audit Capture function that creates information on all auditable events.
- Authorized administration role to allow authorized administrators to configure the policies for discretionary access control, identification and authentication, and auditing. The TOE must enforce the authorized administration role.

57 Security services that must be provided by the TOE or by its IT environment:

- Identification and Authentication (I&A) by which users are uniquely identified and authenticated before they are authorized to access information stored on the DBMS.

- Audit Storage service that stores records for all security-relevant operations that users perform on user and DBMS data.

- Audit Review service that allows the authorized administrator to review stored audit records in order to detect potential and actual security violations.

58 Security services that must be provided by the IT environment:

- Non-bypassibility of the security functions so as to prohibit any access to data or the TOE that is not governed by the TOE security policies.

- Domain separation to ensure that other software operating on the same computer as the TOE cannot interfere with or negate the security functions of the TOE. Domain separation also ensures that multiple instances of the TOE concurrently executing cannot interfere with one another.

59 However, a compliant DBMS will not be able to provide the following:

- Physical protection mechanisms and the administrative procedures for using them.

- Mechanisms to ensure the complete availability of the data residing on the DBMS. The DBMS can provide simultaneous access to data to make the data available to more than one person at a given time, and it can enforce DBMS resource allocation limits to prevent users from monopolizing a DBMS service/resource. However, it cannot detect or prevent the unavailability that may occur as a result of a physical or environmental disaster, a storage device failure, or a hacker attack on the underlying operating system. For such threats to availability, the environment must provide the required countermeasures.

- Mechanisms to ensure that users properly secure the data that they retrieve from the DBMS. The security procedures of the organization(s) that use and manage the DBMS must define users' data retrieval, storage, and disposition responsibilities.

- Mechanisms to ensure that authorized administrators wisely use DAC. Although the DBMS can support an access control policy by which users are granted access only to the data that they need to perform their jobs, it cannot completely ensure that authorized administrators who are able to set access controls will do so prudently.

## 2.4 TOE Operational Environment

### 2.4.1 Basic-Robustness Environment

60 The TOE described in this PP is intended to operate in environments having a basic level of robustness as defined in the Glossary in section 1.4.

61 Basic robustness allows processing of data at a single sensitivity level in an environment where users are cooperative and threats are minimum. Authorized users of the TOE are cleared for all information managed by the DBMS, but may not have the need-to-know

authorization for all of the data. Hence, the risk that significant damage will be done due to compromise of data is low.

62 Entities in the IT environment on which the TOE depends for security functions must be of at least the same level of robustness as the TOE. It is necessary for such an environment that the underlying operating system on which the DBMS is installed be evaluated against a basic robustness protection profile for operating systems.

63 The TOE in and of itself is not of sufficient robustness to store and protect information of such criticality that the integrity or secrecy is critical to the survival of the enterprise.

### 2.4.2 Enclave

64 The term, "enclave", further characterizes the environment in which the TOE is intended to operate. An enclave is under the control of a single authority and has a homogeneous security policy, including personnel and physical security, to protect it from other environments. An enclave can be specific to an organization or a mission and it may contain multiple networks. Enclaves may be logical, such as an operational area network, or be based on physical location and proximity. Any local and external elements that access resources within the enclave must satisfy the policy of the enclave.

65 The DBMS is expected to interact with other IT products that reside in the host OS, in the IT environment in which the host computer and host OS reside, outside that environment but inside the enclave. The IT and non-IT mechanisms used for secure exchanges of information between the DBMS and such products are expected to be administratively determined and coordinated. Similarly, the IT and non-IT mechanisms for negotiating or translating the DAC policy involved in such exchanges are expected to be resolved by the organizations involved.

### 2.4.3 TOE Architectures

66 This PP does not dictate a specific architecture. A TOE may operate in several architectures, for example:

- A stand-alone system running the DBMS server application
- A stand-alone system running the DBMS server and DBMS client(s) and serving one online user at a given time.
- A network of systems communicating with several distributed DBMS servers simultaneously;
- A network of workstations or terminals running DBMS clients and communicating with a DBMS server simultaneously; these devices may be hardwired to the host computer or be connected to it by means of local or wide-area networks.
- A network of workstations communicating with one or more application servers, which in turn interact with the DBMS on behalf of the workstation users or other

subjects (e.g., a DBMS server interacting with a transaction processor that manages user requests).

- A network of workstations communicating with several distributed DBMS servers simultaneously; the DBMS servers may all be within a single local area network, or they may be distributed geographically.

67 This PP allows each of these architectures to be supported as well as others. Figure 3 depicts one of the possible architectures and shows an enclave in which DBMS users access the TOE via a local area network (LAN) and also possibly using a dial-up connection. Users in other enclaves will access the LAN and the host computers and servers on it by way of one or more boundary protection mechanisms (e.g., a firewall) and then through a communications server or router to the LAN. Depending on the particular enclave configuration and the DBMS access policy that it supports, all users (both inside and outside the enclave) may then access an application server, which either connects the TOE user to the enclave computer on which the TOE operates or manages the complete user/DBMS session.

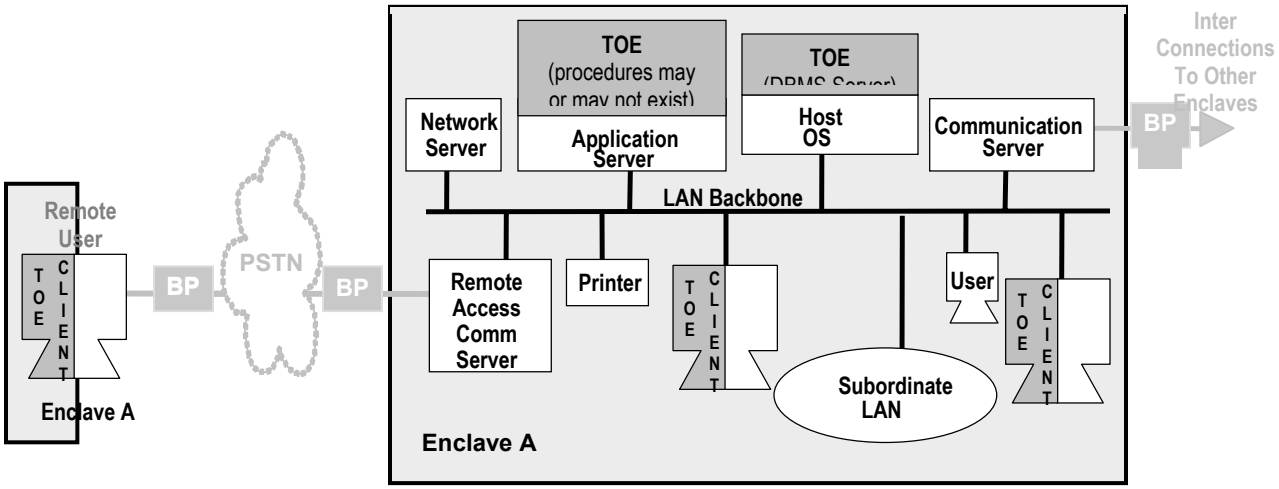



**Figure 3 - Depiction of TOE Configuration**

### 2.4.4 TOE Administration

68 Authorized administrators of the TOE will have capabilities that are commensurate with their assigned administrative roles. There may be one or more administrative roles. The TOE developers will establish some roles for their products. If the security target allows it, the administrators of the system may establish other roles. This PP defines one necessary administrator role (authorized administrator) and allows the DBMS developer or ST writer to define more. When the DBMS is established, the ability to segment roles and assign capabilities with significant freedom regarding the number of roles and their responsibilities

must also exist. Of course the very ability to establish and assign roles will be a privileged function.

# 3 TOE Security Environment

69 The security environment for the functions addressed by this specification includes threats, security policies, and usage assumptions, as discussed below.

## 3.1 Use of basic robustness

70 Basic robustness TOEs falls in the upper left area of the previously discussed robustness figures (Figures 1 and 2). A Basic Robustness TOE is considered sufficient for low threat environments or where compromise of protected information will not have a significant impact on mission objectives. This implies that the motivation of the threat agents will be low in environments that are suitable for TOEs of this robustness. In general, basic robustness results in "good commercial practices" that counter threats based in casual and accidental disclosure or compromise of data protected by the TOE.

71 Threat agent motivation can be considered in a variety of ways. One possibility is that the value of the data processed or protected by the TOE will generally be seen as of little value to the adversary (i.e., compromise will have little or no impact on mission objectives). Another possibility, (where higher value data is processed or protected by the TOE) is that procuring organizations will provide other controls or safeguards (i.e., controls that the TOE itself does not enforce) in the fielded system in order to increase the threat agent motivation level for compromise beyond a level of what is considered reasonable or expected to be applied.

## 3.2 Threat agent characterization

72 In addition to helping define the robustness appropriate for a given environment, the threat agent is a key component of the formal threat statements in the PP. Threat agents are typically characterized by a number of factors such as *motivation, expertise*, and *available resources*. Because each robustness level is associated with a variety of environments, there are corresponding varieties of specific threat agents (that is, the threat agents will have different combinations of motivation, expertise, and available resources) that are valid for a given level of robustness. The following discussion explores the impact of each of the threat agent factors on the ability of the TOE to protect itself (that is, the robustness required of the TOE).

73 The *motivation* of the threat agent seems to be the primary factor of the three characteristics of threat agents outlined above. Given the same expertise and set of resources, an attacker with low motivation may not be as likely to attempt to compromise the TOE. For example, an entity with no authorization to low value data none-the-less has low motivation to compromise the data; thus a basic robustness TOE should offer sufficient protection. Likewise, the fully authorized user with access to highly valued data similarly has low motivation to attempt to compromise the data, thus again a basic robustness TOE should be sufficient.

74 Unlike the motivation factor, however, the same can't be said for *expertise*. A threat agent with low motivation and low expertise is just as unlikely to attempt to compromise a TOE as an attacker with low motivation and high expertise; this is because the attacker with high expertise does not have the motivation to compromise the TOE even though they may have the expertise to do so. The same argument can be made for *resources* as well.

75 Therefore, when assessing the robustness needed for a TOE, the motivation of threat agents should be considered a "high water mark". *That is, the robustness of the TOE should increase as the motivation of the threat agents increases.*

76 Having said that, the relationship between expertise and resources is somewhat more complicated. In general, if resources include factors other than just raw processing power (money, for example), then expertise should be considered to be at the same "level" (low, medium, high, for example) as the resources because money can be used to purchase expertise. Expertise in some ways is different, because expertise in and of itself does not automatically procure resources. However, it may be plausible that someone with high expertise can procure the requisite amount of resources by virtue of that expertise (for example, hacking into a bank to obtain money in order to obtain other resources).

77 It may not make sense to distinguish between these two factors; in general, it appears that the only effect these may have is to lower the robustness requirements. For instance, suppose an organization determines that, because of the value of the resources processed by the TOE and the trustworthiness of the entities that can access the TOE, the motivation of those entities would be "medium". This normally indicates that a medium robustness TOE would be required because the likelihood that those entities would attempt to compromise the TOE to get at those resources is in the "medium" range. However, now suppose the organization determines that the entities (threat agents) that are the least trustworthy have no resources and are unsophisticated. In this case, even though those threat agents have medium motivation, the likelihood that they would be able to mount a successful attack on the TOE would be low, and so a basic robustness TOE may be sufficient to counter that threat.

78 It should be clear from this discussion that there is no "cookbook" or mathematical answer to the question of how to specify exactly the level of motivation, the amount of resources, and the degree of expertise for a threat agent so that the robustness level of TOEs facing those threat agents can be rigorously determined. However, an organization can look at combinations of these factors and obtain a good understanding of the likelihood of a successful attack being attempted against the TOE. Each organization wishing to procure a TOE must look at the threat factors applicable to their environment; discuss the issues raised in the previous paragraph; consult with appropriate accreditation authorities for input; and document their decision regarding likely threat agents in their environment.

79 The important general points we can make are:

- The motivation for the threat agent defines the upper bound with respect to the level of robustness required for the TOE.

- A threat agent's expertise and/or resources that is "lower" than the threat agent's motivation (e.g., a threat agent with high motivation but little expertise and few resources) may lessen the robustness requirements for the TOE (see next point, however).

- The availability of attacks associated with high expertise and/or high availability of resources (for example, via the Internet or "hacker chat rooms") introduces a problem when trying to define the expertise of, or resources available to, a threat agent.

## 3.3 Threats

The following are the threat statements that the TOE must address.

**Table 3 - Basic Robustness Applicable Threats**

| | |
|---|---|
| T. ADMIN_ERROR | A authorized administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms. |
| T.AUDIT_COMPROMISE | A process or user may cause audit data to be inappropriately accessed (viewed, modified or deleted), or prevent future records from being recorded, thus masking an attacker's actions. |
| T.IMPROPER_INSTALLATION | The TOE may be delivered, installed, or initially configured in a manner that undermines TOE security. |
| T.INSECURE_START | Reboot may result in insecure state of the TOE. |
| T.MASQUERADE | An unauthorized user, process, or external IT entity may masquerade as an authorized entity to gain access to data or TOE resources. |
| T.POOR_DESIGN | Unintentional or intentional errors in requirement specification, design or development of the TOE may occur. |
| T.POOR_IMPLEMENTATION | Unintentional or intentional errors in implementing the design of the TOE may occur. |

| | |
|---|---|
| T.POOR_TEST | Lack of or insufficient tests to demonstrate that all TOE security functions operate correctly (including in a fielded TOE) may result in incorrect TOE behavior being undiscovered thereby causing potential security vulnerabilities. |
| T.RESIDUAL_DATA | A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another. |
| T.SYSACC | A malicious process or user may gain unauthorized access to the authorized administrator account, or that of other trusted personnel. |
| T.TSF_COMPROMISE | A malicious user or process may cause configuration data to be inappropriately accessed (viewed, modified or deleted). |
| T.UNATTENDED_SESSION | A user may gain unauthorized access to an unattended session. |
| T.UNAUTH_ACCESS | A user may gain unauthorized access (view, modify, delete) to user data. |
| T.UNDETECTED_ACTIONS | Failure of the IT operating system to detect and record unauthorized actions may occur. |
| T.UNIDENTIFIED_ACTIONS | Failure of the authorized administrator to identify and act upon unauthorized actions may occur. |

**Table 4 - Basic Robustness Threats NOT Applicable**

| Threat Name | Threat Definition | Rationale for NOT Including this Threat |
|---|---|---|
| T.ACCIDENTAL_AUDIT_COMPROMISE | A user or process may view audit records, cause audit records to be lost or modified, or prevent future audit records from being recorded, thus masking a user's action. | The protection profile addresses this with the threat T.AUDIT_COMPROMISE, which could be an accidental compromise or malicious compromise. |

| | | |
|---|---|---|
| T.ACCIDENTAL_ CRYPTO_ COMPROMISE | A user or process may cause key, data or executable code associated with the cryptographic functionality to be inappropriately accessed (viewed, modified, or deleted), thus compromising the cryptographic mechanisms and the data protected by those mechanisms. | This threat is not applicable to the TOE due to the absence of cryptographic requirements for the TOE. |

## 3.4 Organizational Security Policy

80 The following are the policy statements whose enforcement must be provided by the TOE itself or by the IT environment or by some combination of the TOE and the IT environment.

**Table 5 - Security Policy**

| | |
|---|---|
| P.ACCOUNTABILITY | The users of the TOE shall be held accountable for their actions within the TOE. |
| P.AUTHORIZATION | The TOE shall limit the extent of each user's abilities in accordance with the TSP. |
| P.AUTHORIZED_USERS | Access controls will ensure that only those users who have been authorized to access the protected information within the TOE will be able to do so. |
| P.I_AND_A | All users must be identified and authenticated prior to accessing any controlled resources with the exception of public objects. |
| P.INDEPENDENT_TESTING | The TOE must undergo independent testing as part of an independent vulnerability analysis. |

| | |
|---|---|
| P.NEED_TO_KNOW | The TOE must limit the access to information in protected resources to those authorized users who have a need to know that information. |
| P.REMOTE_ADMIN_ACCESS | Authorized administrators shall be able to remotely manage the TOE. |
| P.ROLES | The TOE shall provide an authorized administrator role for secure administration of the TOE. This role shall be separate and distinct from other authorized users. |

## 3.5 Security Usage Assumptions

81 The following table lists the assumptions made about the use of the TOE.

**Table 6 - Security Usage Assumptions**

| | |
|---|---|
| A.NO_EVIL | Authorized administrators are non-hostile, appropriately trained and follow all administrator guidance. |
| A.NO_GENERAL_PURPOSE | There are no general-purpose computing capabilities (e.g., compilers or user applications) available on DBMS servers, other than those services necessary for the operation, administration and support of the DBMS. |
| A.PHYSICAL | It is assumed that appropriate physical security is provided within the domain for the value of the IT assets protected by the TOE and the value of the stored, processed, and transmitted information. |
| A.ROBUST_ENVIRONMENT | It is assumed that the IT environment is at least as robust as the TOE. |
| A.SECURE_COMMS | It is assumed that the IT environment will provide a secure line of communications between the remote user and the TOE. |

# 4 Security Objectives

82 This section defines the security objectives for the TOE and its environment. These objectives are suitable to counter all identified threats and cover all identified organizational security policies and assumptions. The TOE security objectives are identified with "O." appended to the beginning of the name and the environment objectives are identified with "OE." appended to the beginning of the name.

## 4.1 TOE Security Objectives

**Table 7 - TOE Objectives**

| | |
|---|---|
| O.ACCESS | The TOE will ensure that users gain only authorized access to it and to the resources that it controls. |
| O.ADMIN_GUIDANCE | The TOE will provide authorized administrators with the necessary information for secure management of the TOE. |
| O.ADMIN_ROLE | The TOE will provide authorized administrator roles to isolate administrative actions. |
| O.AUDIT_GENERATION | The TOE will provide the capability to detect and create records of security relevant events associated with users. |
| O. AUDIT_PROTECTION | The TOE will provide the capability to protect audit information. |
| O. AUDIT_REVIEW | The TOE will provide the capability to selectively view audit information, and alert the authorized administrator of identified potential security violations. |
| O.DISCRETIONARY_ACCESS | The TOE will control access to resources based upon the identity of users or groups of users. |
| O.INSTALL | The TOE will be delivered with the appropriate installation guidance to establish and maintain TOE security. |

| | |
|---|---|
| O.MANAGE | The TOE will provide all the functions and facilities necessary to support the authorized administrators in their management of the security of the TOE. |
| O.INTERNAL_TOE_DOMAINS | The TSF will maintain internal domains for separation of data and queries belonging to concurrent users. |
| O.PROTECT | The TOE will provide mechanisms to protect user data and resources. |
| O.RESIDUAL_INFORMATION | The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated. |
| O.SOUND_DESIGN | The TOE will be designed using sound design principles and techniques. The TOE design, design principles and design techniques will be adequately and accurately documented. |
| O.SOUND_IMPLEMENTATION | The implementation of the TOE will be an accurate instantiation of its design. |
| O.TESTING | The TOE will undergo developer and independent testing that includes test scenarios and results. |
| O.TRAINED_USERS | The TOE will provide authorized users with the necessary guidance for secure use of the TOE, to include secure sharing of user data. |
| O.USER_AUTHENTICATION | The TOE will verify the claimed identity of users. |
| O.USER_IDENTIFICATION | The TOE will uniquely identify users. |

## 4.2 Environment Security Objectives

**Table 8 - Objectives for the IT Environment**

| | |
|---|---|
| OE.NO_EVIL | Sites using the TOE shall ensure that authorized administrators are non-hostile, appropriately trained and follow all administrator guidance. |

| | |
|---|---|
| OE.CONFIG | The TOE will be installed, configured, managed and maintained in accordance with its guidance documentation and applicable security policies and procedures |
| OE. NO_GENERAL_ PURPOSE | There will be no general-purpose computing capabilities (e.g., compilers or user applications) available on DMBS servers, other than those services necessary for the operation, administration and support of the DBMS. |
| OE.PHYSICAL | Physical security will be provided within the domain for the value of the IT assets protected by the TOE and the value of the stored, processed, and transmitted information. |
| OE.ROBUST_ENVIORNMENT | The IT environment that supports the TOE for enforcement of its security objectives will be of at least the same level of robustness as the TOE. |
| OE.SECURE_COMMS | The IT environment will provide a secure line of communications between the remote user and the TOE. |
| OE.SELF_PROTECTION | IT environment and its assets will be protected from external interference, tampering or unauthorized disclosure. |
| OE.TOE_PROTECTION | The IT environment will provide protection to the TOE and its assets from external interference or tampering. |
| OE.TIME | The IT environment will provide a time source that provides reliable time stamps. |
| OE.TRUST_IT | Each IT entity the TOE relies on for security functions will be installed, configured, managed and maintained in a manner appropriate to the IT entity, and consistent with the security policy of the TOE and the relationship between them. |

# 5 IT Security Requirements

83 This section provides functional and assurance requirements that must be satisfied by a PP-compliant TOE. These requirements consist of functional components from Part 2 of the CC and assurance components from Part 3 of the CC in the form of EAL 2 augmented.

84 This protection profile is current with NIAP and International Common Criteria interpretations as of 24 January 2003. All interpretations in this profile are expressed as explicit requirements. There are two interpretations that were consciously not adopted:

- NIAP I-0407 – The authors believe this interpretation is not necessary.
- RI# 65 – The authors believe this interpretation creates a requirement that is redundant with other requirements in the document.

**Table 9 - TOE Security Functional Requirements**

| Functional Class | Functional Components | |
|---|---|---|
| Class FAU: Security Audit | FAU_GEN_EXP.1 | Audit data generation |
| | FAU_GEN_EXP.2 | User identity association |
| | FAU_SAR.1 | Audit review (TOE) |
| | FAU_SAR.2 | Restricted Audit Review |
| | FAU_SAR.3 | Selectable audit review (TOE) |
| | FAU_SEL.1 | Selective audit |
| | FAU_STG_EXP.1 | Protected audit trail storage (TOE) |
| | FAU_STG_EXP.2 | Site-configurable Prevention of audit data loss |
| Class FDP: User Data Protection | FDP_ACC.1 | Subset access control |
| | FDP_ACF.1 | Security attribute based access control |
| | FDP_ITT.1 | Basic internal transfer protection |
| | FDP_RIP.2 | Full residual information protection |
| Class FIA: Identification and Authentication | FIA_AFL_EXP.1 | Authentication failure handling |
| | FIA_ATD.1 | User attribute definition |
| | FIA_SOS.1 | Verification of secrets |
| | FIA_UAU.1 | Timing of authentication (TOE) |
| | FIA_UID.1 | Timing of identification (TOE) |
| | FIA_USB_EXP.1 | User-subject binding |

| Functional Class | Functional Components | |
|---|---|---|
| Class FMT: Security Management | FMT_MOF.1 | Management of security functions behavior |
| | FMT_MSA.1 | Management of DAC security attributes |
| | FMT_MSA.2 | Secure security attributes |
| | FMT_MSA.3 | Static attribute initialization |
| | FMT_MTD.1(1) | Management of TSF data (audit events) |
| | FMT_MTD.1(2) | Management of TSF data (audit records) |
| | FMT_MTD.1(3) | Management of TSF data (user authentication data) |
| | FMT_REV.1(1) | Revocation (user attributes) |
| | FMT_REV.1(2) | Revocation (subject, object attributes) |
| | FMT_SMR.1 | Security roles |
| Class FPT: Protection of the TOE Security Functions | FPT_ITD_EXP.1 | Internal TOE domains |
| | FPT_RVM.1 | Non-bypass ability of the TSP |
| | FPT_ITT.1 | Basic Internal TSF Data Transfer Protection |
| | FPT_TRC_EXP.1 | Explicit: Internal TSF consistency |
| Class FTA: TOE Access | FTA_MCS.1 | Basic limitation on multiple concurrent sessions |
| | FTA_TSE.1 | TOE session establishment |

# 5.1 TOE Security Functional Requirements

85 This section contains the security functional requirements that must be satisfied by the TOE.

## 5.1.1 Security audit (FAU)

### 5.1.1.1 Explicit: Audit data generation (FAU_GEN_EXP.1)

FAU_GEN_EXP.1.1 **Refinement:** The TSF shall be able to generate an audit record of the following auditable events:

a) Start-up and shutdown of the audit functions;

b) All auditable events, **listed in Table 10,** for the minimum level of audit;

**c) Start-up and shutdown of the DBMS;**

**d) Use of special permissions (e.g., those often used by authorized administrators to circumvent access control policies)**; and

e) *[assignment: other specifically defined auditable events.].*

FAU_GEN_EXP.1.2 **Refinement:** The TSF shall record within each audit record at least the following information:

a) Date and time of the event, type of event, subject identity (if applicable), and the outcome (success or failure) of the event;

b) For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST: **For changes to TSF data, except for authentication data, the new and old value of the data;** and

*c)* For each audit event type, based on the auditable event definitions of the functional components included in the PP/ST *[assignment: other audit relevant information, excluding sensitive fields]*

Dependencies:

FPT_STM.1 Reliable time stamps

**Table 10 - FAU_GEN.1 Auditable events**

| **Security Functional Requirement** | **Auditable Event(s)** | **Additional Audit Record Contents** |
|---|---|---|
| FAU_SAR.1<br>Audit review | None | |
| FAU_SAR.2<br>Restricted Audit Review | None | |
| FAU_SAR.3<br>Selectable Audit Review | None | |
| FAU_SEL.1<br>Selective audit | All modifications to the audit configuration that occur while the audit collection functions are operating. | The identity of the authorized administrator that made the change to the audit configuration. |
| FAU_STG_EXP.1<br>Protected audit trail storage | None | |
| FAU_STG_EXP.2<br>Site-configurable Prevention of audit data loss | Actions taken due to the audit storage failure.<br>Selection of an action to be taken when there is an audit storage failure. | The identity of the authorized administrator selecting the action to be taken in case of audit storage failure. |
| FDP_ACC.1<br>Subset access control | None | |
| FDP_ACF.1<br>Security attribute based access control | Successful requests to perform an operation on an object covered by the SFP. | The identity of the subject performing the operation. |
| FDP_ITT.1<br>Basic internal transfer protection | Successful transfers of user data, including identification of the protection method used. | Identity of the individual transferring data.<br>Identity of authorized administrator attempting to change the integrity protection method. |
| FDP_RIP.2<br>Full residual information protection | None | |

| Security Functional Requirement | Auditable Event(s) | Additional Audit Record Contents |
|---|---|---|
| FIA_AFL_EXP.1<br>Authentication failure handling | The reaching of the threshold for the unsuccessful authentication attempts and the actions (e.g., disabling of a terminal) taken and the subsequent, if appropriate, restoration to the normal state (e.g., re-enabling of a terminal). | |
| FIA_ATD.1<br>User attribute definition | None | |
| FIA_SOS.1<br>Verification of secrets | Rejection by the TSF of any tested secret. | |
| FIA_UAU.1<br>Timing of authentication | Unsuccessful use of the authentication mechanism. | Identity of the user or authorized administrator that entered the incorrect authentication data, but not the incorrect authentication data itself. |
| FIA_UID.1<br>Timing of identification | Unsuccessful use of the user identification mechanism, including the user identity provided. | Identification information entered. |
| FIA_USB_EXP.1<br>User-subject binding | Unsuccessful binding of user security attributes to a subject (e.g., creation a subject). | Identity of subject, and user security attributes, except private attributes (e.g., private key) |
| FMT_MOF.1<br>Management of security functions behavior | None | |
| FMT_MSA.1<br>Management of security attributes | None | |
| FMT_MSA.2<br>Secure security attributes | All offered and rejected values for a security attribute. | |
| FMT_MSA.3<br>Static attribute initialization | None | |
| FMT_MTD.1<br>Management of TSF data | None | |
| FMT_REV.1<br>Revocation | Unsuccessful revocation of security attributes. | Identity of individual attempting to revoke security attributes. |
| FMT_SMR.1<br>Security roles | Modifications to the users that are part of a role. | Identity of authorized administrator modifying the role definition |

| **Security Functional Requirement** | **Auditable Event(s)** | **Additional Audit Record Contents** |
|---|---|---|
| FPT_RVM.1<br>Non-bypassability of the TSP | None | |
| FPT_ITD_EXP.1<br>Internal TOE domains | None | |
| FPT_ITT.1 | None | |
| FPT_TRC_EXP.1<br>Explicit: Internal TSF consistency | Restoring consistency upon reconnection | |
| FTA_MCS.1<br>Basic limitation on multiple concurrent sessions | Rejection of a new session based on the limitation of multiple concurrent sessions. | |
| FTA_TSE.1<br>TOE session establishment | Denial of a session establishment due to the session establishment mechanism. | Identity of the individual attempting to establish a session |

#### 5.1.1.2 Explicit: User identity association (FAU_GEN_EXP.2)

**FAU_GEN_EXP.2.1** For audit events resulting from actions of identified users, the TSF shall be able to associate each auditable event with the identity of the user that caused the event.

Dependencies:

FAU_GEN.1 Audit data generation

FIA_UID.1 Timing of identification

#### 5.1.1.3 Selective audit (FAU_SEL.1)

FAU_SEL.1.1The TSF shall be able to include or exclude auditable events from the set of audited events based on the following attributes:

a) object identity;

b) user identity;

c) event type;

***d)*** **success of auditable security events;**

***e)*** **failure of auditable security events;**

f) *[assignment: list of additional attributes that audit selectivity is based upon].*

Dependencies:

FAU_GEN.1 Audit data generation

FMT_MTD.1 Management of TSF data

## 5.1.2 User data protection (FDP)

### 5.1.2.1 Subset access control (FDP_ACC.1)

FDP_ACC.1.1 The TSF shall enforce the **Discretionary Access Control policy** on **all subjects, all DBMS-controlled objects and all operations among them.**

Dependencies:

FDP_ACF.1 Security attribute based access control

### 5.1.2.2 Security attribute based access control (FDP_ACF.1)

FDP_ACF.1.1 The TSF shall enforce the **Discretionary Access Control policy** to objects based on

**a) the authorized user identity associated with a subject, and**

**b) access operations implemented for DBMS-controlled objects.**

86 *Application Note: DBMS-controlled objects may be implementation-specific objects that are presented to authorized users at the user interface to the DBMS. They may include, but are not limited to tables, records, files, indexes, views, constraints, stored queries, and metadata. Data structures that are not presented to authorized users at the DBMS user interface, but are used internally are internal TSF data structures. Internal TSF data structures are not controlled according to the rules specified in FDP_ACF.1.*

FDP_ACF.1.2 **Refinement:** The TSF shall enforce the following rules to determine if an operation among subjects and **DBMS**-controlled objects is allowed:**1**

- **The Discretionary Access Control policy mechanism shall, either by explicit authorized user action or by default, provide that database management system controlled objects are protected from unauthorized access according to the following ordered rules:**

  **1) If the requested mode of access is denied to that authorized user, deny access.**

**2) If the requested mode of access is permitted to that authorized user, permit access.**

3) **Else deny access.**

FDP_ACF.1.3 **Refinement:** The TSF shall explicitly authorize access of subjects to **DBMS-controlled** objects based on the following additional rules: **Authorized administrators must follow the above-stated Discretionary Access Control policy, except after taking the following specific actions: *[assignment: list of specific actions]*.**

87 *Application Note: This element allows specifications of additional rules for authorized administrators to bypass the Discretionary Access Control policy for system management or maintenance (e.g., system backup).*

FDP_ACF.1.4 The TSF shall explicitly deny access of subjects to objects based on the *[assignment: rules, based on security attributes, that explicitly deny access of subjects to objects]*

Dependencies:

FDP_ACC.1 Subset access control

FMT_MSA.3 Static attribute initialization

### 5.1.2.3 Basic internal transfer protection (FDP_ITT.1)

FDP_ITT.1.1 The TSF shall enforce the **Discretionary Access Control policy** to prevent the disclosure of user data when it is transmitted between physically-separated parts of the TOE.

Dependencies:

[FDP_ACC.1 Subset access control, or

FDP_IFC.1 Subset information flow control]

### 5.1.2.4 Full residual information protection (FDP_RIP.2)

FDP_RIP.2.1 The TSF shall ensure that any previous information content of a resource is made unavailable upon the allocation of the resource to all objects.

Dependencies: No dependencies

### 5.1.3 Identification and authentication (FIA)

#### 5.1.3.1 User attribute definition (FIA_ATD.1)

FIA_ATD.1.1 The TSF shall maintain the following list of security attributes belonging to individual users:

a) **Database user identifier;**

b) **Security-relevant database roles;** and

*c) [assignment: list of security attributes]*

Dependencies: No dependencies

88 .

### 5.1.4 Security management (FMT)

#### 5.1.4.1 Management of security functions behavior (FMT_MOF.1)

FMT_MOF.1.1 The TSF shall restrict the ability to disable and enable the functions **relating to the specification of events to be audited** to **authorized administrators**.

Dependencies:

FMT_SMR.1 Security roles

#### 5.1.4.2 Management of security attributes (FMT_MSA.1)

FMT_MSA.1.1 **Refinement:** The TSF shall enforce the **Discretionary Access Control policy** to restrict the ability to **manage** the security attributes **of database users** to **authorized administrators**.

Dependencies:

[FDP_ACC.1 Subset access control or

FDP_IFC.1 Subset information flow control]

FMT_SMR.1 Security roles

89 *Application Note: The ST author should ensure that all attributes identified in FDP_ACF.1 are adequately managed and protected.*

90

#### 5.1.4.3 Secure security attributes (FMT_MSA.2)

FMT_MSA.2.1 The TSF shall ensure that only secure values are accepted for security attributes.

Dependencies:

ADV_SPM.1 Informal TOE security policy model

[FDP_ACC.1 Subset access control or

FDP_IFC.1 Subset information flow control]

FMT_MSA.1 Management of security attributes

FMT_SMR.1 Security roles

91 *Application note: The TOE's security policy model, along with the resulting information in the guidance document, should address the definition of 'secure' as used in this requirement.*

#### 5.1.4.4 Static attribute initialization (FMT_MSA.3)

FMT_MSA.3.1 The TSF shall enforce the **Discretionary Access Control policy** to provide restrictive default values for security attributes that are used to enforce the SFP.

FMT_MSA.3.2 **Refinement:** The TSF shall allow the **authorized administrator** to specify alternative initial values to override the default values when an object is created.

Dependencies:

FMT_MSA.1 Management of security attributes

FMT_SMR.1 Security roles

#### 5.1.4.5 Management of TSF data (audit events) (FMT_MTD.1(1))

FMT_MTD.1.1(1) The TSF shall restrict the ability to include or exclude the **auditable events** to **authorized administrators**.

Dependencies:

FMT_SMR.1 Security roles

#### 5.1.4.6 Revocation (FMT_REV.1(1))

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the users within the TSC to **the authorized administrator**.

FMT_REV.1.2 The TSF shall enforce the rules *[assignment: specification of revocation rules].*

Dependencies:

FMT_SMR.1 Security roles

#### 5.1.4.7 Revocation (FMT_REV.1(2))

FMT_REV.1.1 The TSF shall restrict the ability to revoke security attributes associated with the subjects and objects within the TSC to **the authorized administrator and database users as allowed by the Discretionary Access Control policy**.

FMT_REV.1.2 The TSF shall enforce the rules *[assignment: specification of revocation rules].*

Dependencies:

FMT_SMR.1 Security roles

#### 5.1.4.8 Security roles (FMT_SMR.1)

FMT_SMR.1.1 The TSF shall maintain the roles:

a) **authorized administrator**; and

*b) [assignment: additional authorized identified roles].*

FMT_SMR.1.2 The TSF shall be able to associate users with roles.

Dependencies:

FIA_UID.1 Timing of identification

92 *Application Note: This requirement identifies a minimum set of management roles. A ST or operational environment may contain a finer-grain decomposition of roles that correspond to the roles identified here (e.g., database non-administrative user or database operator). The ST writer may change the names of the roles identified above but the "new" roles must still perform the functions that the FMT requirements in this PP have defined.*

### 5.1.5 Protection of the TOE Security Functions (FPT)

#### 5.1.5.1 Explicit: SFP domain separation (FPT_ITD_EXP.1)

**FPT_ITD_EXP.1.1** The TSF shall enforce separation between the security domains of subjects in the TOE Scope of Control.

Dependencies: No dependencies

#### 5.1.5.2 Basic Internal TSF Data Transfer Protection (FPT_ITT.1)

FPT_ITT.1.1 The TSF shall protect TSF data from disclosure and modification when it is transmitted between separate parts of the TOE.

Dependencies: No dependencies

#### 5.1.5.3 Non-bypassability of the TSP (FPT_RVM.1)

FPT_RVM.1.1 The TSF shall ensure that TSP enforcement functions are invoked and succeed before each function within the TSC is allowed to proceed.

Dependencies: No dependencies

#### 5.1.5.4 Explicit: Internal TSF consistency (FPT_TRC_EXP.1)

**FPT_TRC_EXP.1.1** The TSF shall ensure that TSF data is consistent between parts of the TOE by providing a mechanism to bring inconsistent TSF data into a consistent state in a timely manner.

93 *Application Note: In general, it is impossible to achieve complete, constant consistency of TSF data that is distributed to remote portions of a TOE because distributed portions of the TSF may be active at different times or disconnected from one another. This requirement attempts to address this situation in a practical manner by acknowledging that there will be TSF data inconsistencies but that they will be corrected without undue delay. For example, a TSF could provide timely consistency through periodic broadcast of TSF data to all TSF nodes maintaining replicated TSF data. Another example approach is for the TSF to provide a mechanism to explicitly probe remote TSF nodes for inconsistencies and respond with action to correct the identified inconsistencies.*

### 5.1.6 Toe Access (FTA)

#### 5.1.6.1 Basic limitation on multiple concurrent sessions (FTA_MCS.1)

FTA_MCS.1.1 The TSF shall restrict the maximum number of concurrent sessions that belong to the same user.

FTA_MCS.1.2 The TSF shall enforce, by default, a limit of [*assignment: default numbe*r] sessions per user.

Dependencies:

FIA_UID.1 Timing of identification

#### 5.1.6.2 TOE session establishment (FTA_TSE.1)

FTA_TSE.1.1 The TSF shall be able to deny session establishment based on **attributes that can be set explicitly by authorized administrator(s) or authorized administrator role(s), including user identity, port of entry, time of day, day of the week, and** *[assignment: list of additional attributes]*.

Dependencies: No dependencies

### 5.1.7 Strength of Function

94 The minimum strength of function level for this protection profile is SOF-basic. See AVA_SOF, Section 5.4.8.2.

## 5.2 Security Requirements for the TOE or the IT Environment

95 This section contains the security functional requirements that must be satisfied either by the TOE or by the IT environment. Security Targets for implementations that use other components in the IT environment to satisfy these requirements must indicate the requirements that are allocated to the IT environment, and do not need to be satisfied by the TOE. In this case, evidence must be provided that the IT environment satisfies these IT functional requirements. This evidence is usually in the form of a completed evaluation of the IT environment component showing that the security functional requirements needed by the DBMS TOE are indeed provided in the IT environment. If sufficient evidence cannot be shown that these requirements are satisfied in the IT environment, it is necessary for the TOE to provide these requirements.

### 5.2.1 Security audit (FAU)

#### 5.2.1.1 Audit review (FAU_SAR.1)

FAU_SAR.1.1 The TSF shall provide **the authorized administrator** with the capability to read **all database audit information** from the audit records.

FAU_SAR.1.2 **Refinement**: The TSF shall provide the audit records in a manner suitable for the **authorized administrator** to interpret the information.**2**

Dependencies:

FAU_GEN.1 Audit data generation

#### 5.2.1.2 Restricted Audit Review (FAU_SAR.2)

FAU_SAR.2.1 The TSF shall prohibit all users read access to the audit records, except those users that have been granted explicit read-access.

Dependencies:

FAU_SAR.1 Audit review

#### 5.2.1.3 Selectable Audit Review (FAU_SAR.3)

FAU_SAR.3.1 The TSF shall provide the ability to perform searches and sorting of audit data based on

a) **User identity;**

b) **Date of event;**

c) **Time of event;**

d) **Type of event;**

e) **Event status (success/failure);** and

*f) [assignment: additional criteria with logical relations]*

Dependencies:

FAU_SAR.1 Audit review

#### 5.2.1.4 Explicit: Protected audit trail storage (FAU_STG_EXP.1)

**FAU_STG_EXP.1.1 Refinement:** The TSF shall **restrict the deletion of** the stored audit records **in the audit trail to the authorized administrator.**[3]

**FAU_STG_EXP.1.2** The TSF shall be able to prevent unauthorized modifications to the audit records in the audit trail.

Dependencies:

FAU_GEN.1 Audit data generation

96 *Application note: Requirement FAU_STG.1.1 refined according to guidance from the PPRB, Basic Robustness PP Guideline, Recommendation 9, dated 24 July, 2002.*

#### 5.2.1.5 Explicit: Site-configurable Prevention of audit data loss (FAU_STG_EXP.2)

**FAU_STG_EXP.2.1 Refinement:** The TSF shall provide the **authorized admin**istrator the capability to select one or more of the following actions, *[selection: 'ignore auditable events', 'prevent auditable events, except those taken by the **authorized administrator'**, 'overwrite the oldest stored audit records']*, and *[assignment: other actions to be taken in case of audit storage failure]* to be taken if the audit trail is full.[4]

**FAU_STG_EXP.2.2** The TSF shall *[selection: 'ignore auditable events', 'prevent auditable events, except those taken by the authorized user with*

*special rights', 'overwrite the oldest stored audit records', [assignment: other actions to be taken in case of audit storage failure]]* if the audit trail is full and no other action has been selected.

97 *Application Note: The TOE provides the authorized administrator the option of preventing audit data loss by preventing auditable events from occurring. The authorized administrator's actions under these circumstances are not required to be audited. The TOE also provides the authorized administrator the option of overwriting "old" audit records rather than preventing auditable events, which may protect against a denial-of-service attack. A denial of service attack could result if auditable events are not allowed to occur, as the normal operation of the DBMS would cause auditable events, such as users logging into the TOE.*

98 *Application Note: The ST writer should fill in other technology-specific actions that can be taken for audit storage failure (in addition to the two already specified), or select "no additional options" if there are no such technology-specific actions.*

## 5.2.2 Identification and authentication (FIA)

### 5.2.2.1 Explicit: Authentication failure handling (FIA_AFL_EXP.1)

**FIA_AFL_EXP.1.1** The TSF shall detect when an authorized administrator configurable integer of unsuccessful authentication attempts occur related to **all user authentication processes**.

**FIA_AFL_EXP.1.2** When the defined number of unsuccessful authentication attempts has been met or surpassed, the TSF shall **prevent the *[assignment: entities requesting authentication]* from performing activities that require authentication until an action is taken by the authorized administrator**.

99 *Application note: The ST authors should ensure that when the* **entities requesting authentication** *is specified in the ST, at least one account should be exempted from the requirement so as to avoid an administrative denial of service.*

100 *Note the use of "authorized administrator" in this requirement. Since this requirement may be met by the TOE or by a component in the IT environment, it is not possible to specify that the authorized individual be a authorized administrator.*

### 5.2.2.2 Verification of secrets (FIA_SOS.1)

FIA_SOS.1.1 The TSF shall provide a mechanism to verify that secrets meet **the following:**

**a) For each attempt to use the authentication mechanism, the probability that a random attempt will succeed is less than one in $5 \times 10^{15}$;** and

101 *Application Note: This can be achieved with a password of eight characters, assuming an alphabet of 92 characters.*

102 *Application Note: The ST specifies the method of authentication. Where authentication is provided by a password mechanism, the ST shows that the restrictions upon passwords (length, alphabet, and other characteristics) result in a password space conforming to item (a) above. Where authentication is provided by a mechanism other than passwords, the ST shows the authentication method has a low probability that authentication data can be forged or guessed.*

**b) Any feedback given during an attempt to use the authentication mechanism will not reduce the probability below the above metrics.**

#### 5.2.2.3 Timing of authentication (FIA_UAU.1)

FIA_UAU.1.1 The TSF shall allow *[assignment: list of TSF mediated actions]* on behalf of the user to be performed before the user is authenticated.

FAI_UAU.1.2 The TSF shall require each user to be successfully authenticated before allowing any other TSF-mediated actions on behalf of that user.

Dependencies:

FIA_UID.1 Timing of identification

103 *Application Note: FIA_UAU might be satisfied by the TOE or by the external IT environment (e.g., by the host operating system). Security Targets for implementations that use other components to satisfy this requirement should indicate that this requirement is allocated to the IT environment, and does not need to be satisfied by the TOE. In this case, evidence must be provided that the IT environment satisfies this IT functional requirement.*

#### 5.2.2.4 Timing of identification (FIA_UID.1)

FIA_UID.1.1 The TSF shall allow *[assignment: list of TSF-mediated action]* on behalf of the user to be performed before the user is identified*.*

FIA_UID.1.2 **Refinement:** The TSF shall require each user to be **uniquely and** successfully identified before allowing any other TSF-mediated actions on behalf of that user.

Dependencies: No dependencies

104 *Application Note: FIA_UID might be satisfied by the TOE or by the external IT environment (e.g., by the host operating system). Security Targets for implementations that use other components to satisfy this requirement should indicate that this requirement is allocated to the IT environment, and does not need to be satisfied by the TOE. In this case, evidence must be provided that the IT environment satisfies this IT functional requirement.*

#### 5.2.2.5 Explicit: User-subject binding (FIA_USB_EXP.1)

**FIA_USB_EXP.1.1** The TSF shall associate the following user security attributes with subjects acting on behalf of that user:

a) **Database user identifier;**

b) **Security-relevant database roles;** and

*c) [assignment: list of additional user security attributes to be bound].*

Dependencies:

FIA_ATD.1 User attribute definition

### 5.2.3 Security management (FMT)

#### 5.2.3.1 Management of TSF data (audit records) (FMT_MTD.1(2))

FMT_MTD.1.1(2) The TSF shall restrict the ability to query and clear the **audit records** to the **authorized administrator**.

105 *Note the use of "authorized administrator" in this requirement. Since this requirement may be met by the TOE or by a component in the IT environment, it is not possible to specify that the authorized individual be a authorized administrator.*

Dependencies:

FMT_SMR.1 Security roles

#### 5.2.3.2 Management of TSF data (user authentication data)(FMT_MTD.1(3))

FMT_MTD.1.1(3) The TSF shall restrict the ability to **set and reset** the **user authentication data** to the **authorized administrator.**

106 *Note the use of "authorized administrator" in this requirement. Since this requirement may be met by the TOE or by a component in the IT environment, it is not possible to specify that the authorized individual be a authorized administrator.*

Dependencies:

FMT_SMR.1 Security roles

## 5.3 Security Requirements for the IT Environment

107 This section contains the security functional requirements for the IT environment. With the TOE being a software-only TOE, the IT environment necessarily must provide protection of the TOE from tampering and interference.

### 5.3.1 Protection of the TSF (FPT)

#### 5.3.1.1 Non-bypassability of the TSP (FPT_RVM.1)

FPT_RVM.1.1 **Refinement:** The **host OS security functions** shall ensure that **host OS security policy** enforcement functions are invoked and succeed before each function within the **scope of control of the host OS** is allowed to proceed.**5**

Dependencies: No dependencies

#### 5.3.1.2 TSF domain separation (FPT_SEP.1)

FPT_SEP.1.1 **Refinement:** The **security functions of the host OS** shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.**6**

FPT_SEP.1.2 **Refinement:** The **security functions of the host OS** shall enforce separation between the security domains of subjects in the **scope of control of the host OS.7**

Dependencies: No dependencies

#### 5.3.1.3 Reliable time stamps (FPT_STM.1)

FPT_STM.1.1 **Refinement:** The **security functions of the host OS** shall be able to provide reliable time stamps for its own use **and for the TOE.8**

Dependencies: No dependencies

108 *Application note: The TOE referenced in this requirement is the TOE of the DBMS.*

## 5.4 TOE Security Assurance Requirements

109 The security assurance requirements for the TOE are equivalent with the Evaluation Assurance Level 2 (EAL2) requirements augmented from part 3 of the Common Criteria with Flaw Remediation (ALC_FLR.2), Misuse-Examination Guidance (AVA_MSU.1), and Informal Security Policy Modeling (ADV_SPM.1). There is no refinement or iteration of any of the assurance requirements. Table 11 lists the classes, families, and components of the EAL2 assurance requirements augmented.

**Table 11 - Basic Robustness Assurance Requirements**

| Assurance classes | Assurance components | |
|---|---|---|
| Class ACM: Configuration Management | ACM_CAP_EXP.2 | Configuration Items |
| Class ADO: Delivery and Operation | ADO_DEL.1 | Delivery Procedures |
| | ADO_IGS.1 | Installation, Generation, And Start-Up Procedures |

| Assurance classes | Assurance components | |
|---|---|---|
| Class ADV: Development | ADV_FSP.1 | Informal Functional Specification |
| | ADV_HLD.1 | Descriptive High Level Design |
| | ADV_RCR.1 | Informal Correspondence Demonstration |
| | ADV_SPM.1 | Informal TOE security policy model |
| Class AGD: Guidance Documents | AGD_ADM.1 | Administrator Guidance |
| | AGD_USR.1 | User Guidance |
| Class ALC: Life Cycle Support | ALC_FLR.2 | Flaw Reporting Procedures |
| Class AVA: Vulnerability Assessment | AVA_MSU.1 | Examination of Guidance |
| | AVA_SOF.1 | Strength Of TOE Security Function Evaluation |
| | AVA_VLA.1 | Developer Vulnerability Analysis |

### 5.4.1 Configuration management (ACM)

#### 5.4.1.1 Configuration Items (ACM_CAP_EXP.2)

Dependencies: No dependencies.

Developer action elements:

ACM_CAP_EXP.2.1D - The developer shall provide a reference for the TOE.

ACM_CAP_EXP.2.3D - The developer shall provide CM documentation.

110 *Application Note: ACM_CAP.2.2D is deleted per NIAP Interpretation I-0412*

Content and presentation of evidence elements:

ACM_CAP_EXP.2.1.C - The reference for the TOE shall be unique to each version of the TOE.

ACM_CAP_EXP.2.2C - The TOE shall be labeled with its reference.

ACM_CAP_EXP.2.3C - The CM documentation shall include a configuration list.

ACM_CAP_EXP.2.4C - The configuration list shall describe the configuration items that comprise the TOE.

ACM_CAP_EXP.2.5C - The CM documentation shall describe the method used to uniquely identify the configuration items.

ACM_CAP_EXP.2.6C-NIAP-0412 - The configuration list shall uniquely identify all configuration items.

Evaluator action elements:

ACM_CAP_EXP.2.1E - The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

## 5.4.2 Delivery and operation (ADO)

### 5.4.2.1 Delivery Procedures (ADO_DEL.1)

Dependencies: No dependencies.

Developer action elements:

ADO_DEL.1.1D The developer shall document procedures for delivery of the TOE or parts of it to the user.

ADO_DEL.1.2D The developer shall use the delivery procedures.

Content and presentation of evidence elements:

ADO_DEL.1.1C The delivery documentation shall describe all procedures that are necessary to maintain security when distributing versions of the TOE to a user's site.

Evaluator action elements:

ADO_DEL.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### 5.4.2.2 Installation, Generation, And Start-Up Procedures (ADO_IGS.1)

Dependencies:

AGD_ADM.1 Administrator guidance

Developer action elements:

ADO_IGS.1.1D The developer shall document procedures necessary for the secure installation, generation, and start-up of the TOE.

Content and presentation of evidence elements:

ADO_IGS.1.1C The documentation shall describe the steps necessary for secure installation, generation, and start-up of the TOE.

Evaluator action elements:

ADO_IGS.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADO_IGS.1.2E The evaluator shall determine that the installation, generation, and start-up procedures result in a secure configuration.

## 5.4.3 Development (ADV)

### 5.4.3.1 Informal Functional Specification (ADV_FSP.1)

Dependencies:

ADV_RCR.1 Informal correspondence demonstration

Developer action elements:

ADV_FSP.1.1D The developer shall provide a functional specification.

Content and presentation of evidence elements:

ADV_FSP.1.1C The functional specification shall describe the TSF and its external interfaces using an informal style.

ADV_FSP.1.2C The functional specification shall be internally consistent.

ADV_FSP.1.3C The functional specification shall describe the purpose and method of use of all external TSF interfaces, providing details of effects, exceptions and error messages, as appropriate.

ADV_FSP.1.4C The functional specification shall completely represent the TSF.

Evaluator action elements:

ADV_FSP.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_FSP.1.2E The evaluator shall determine that the functional specification is an accurate and complete instantiation of the TOE security functional requirements.

### 5.4.3.2 Descriptive High Level Design (ADV_HLD.1)

Dependencies:

ADV_FSP.1 Informal functional specification

ADV_RCR.1 Informal correspondence demonstration

Developer action elements:

ADV_HLD.1.1D The developer shall provide the high-level design of the TSF.

Content and presentation of evidence elements:

ADV_HLD.1.1C The presentation of the high-level design shall be informal.

ADV_HLD.1.2C The high-level design shall be internally consistent.

ADV_HLD.1.3C The high-level design shall describe the structure of the TSF in terms of subsystems.

ADV_HLD.1.4C The high-level design shall describe the security functionality provided by each subsystem of the TSF.

ADV_HLD.1.5C The high-level design shall identify any underlying hardware, firmware, and/or software required by the TSF with a presentation of the functions provided by the supporting protection mechanisms implemented in that hardware, firmware, or software.

ADV_HLD.1.6C The high-level design shall identify all interfaces to the subsystems of the TSF.

ADV_HLD.1.7C The high-level design shall identify which of the interfaces to the subsystems of the TSF are externally visible.

Evaluator action elements:

ADV_HLD.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ADV_HLD.1.2E The evaluator shall determine that the high-level design is an accurate and complete instantiation of the TOE security functional requirements.

#### 5.4.3.3 Informal Correspondence Demonstration (ADV_RCR.1)

Dependencies: No dependencies.

Developer action elements:

ADV_RCR.1.1D The developer shall provide an analysis of correspondence between all adjacent pairs of TSF representations that are provided.

Content and presentation of evidence elements:

ADV_RCR.1.1C For each adjacent pair of provided TSF representations, the analysis shall demonstrate that all relevant security functionality of the more abstract TSF representation is correctly and completely refined in the less abstract TSF representation.

Evaluator action elements:

ADV_RCR.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.4.3.4 Informal TOE security policy model (ADV_SPM.1)

Dependencies:

ADV_FSP.1 Informal functional specification

Developer action elements:

ADV_SPM.1.1D The developer shall provide a TSP model.

ADV_SPM.1.2D The developer shall demonstrate correspondence between the functional specification and the TSP model.

Content and presentation of evidence elements:

ADV_SPM.1.1C The TSP model shall be informal.

ADV_SPM.1.2C The TSP model shall describe the rules and characteristics of all policies of the TSP that can be modeled.

ADV_SPM.1.3C The TSP model shall include a rationale that demonstrates that it is consistent and complete with respect to all policies of the TSP that can be modeled.

ADV_SPM.1.4C The demonstration of correspondence between the TSP model and the functional specification shall show that all of the security functions in the functional specification are consistent and complete with respect to the TSP model.

Evaluator action elements:

ADV_SPM.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### 5.4.4 Guidance documents (AGD)

#### 5.4.4.1 Administrator Guidance (AGD_ADM.1)

Dependencies:

ADV_FSP.1 Informal functional specification

Developer action elements:

AGD_ADM.1.1D The developer shall provide administrator guidance addressed to system administrative personnel.

Content and presentation of evidence elements:

AGD_ADM.1.1C The administrator guidance shall describe the administrative functions and interfaces available to the administrator of the TOE.

AGD_ADM.1.2C The administrator guidance shall describe how to administer the TOE in a secure manner.

AGD_ADM.1.3C The administrator guidance shall contain warnings about functions and privileges that should be controlled in a secure processing environment.

AGD_ADM.1.4C The administrator guidance shall describe all assumptions regarding user behavior that are relevant to secure operation of the TOE.

AGD_ADM.1.5C The administrator guidance shall describe all security parameters under the control of the administrator, indicating secure values as appropriate.

AGD_ADM.1.6C The administrator guidance shall describe each type of security-relevant event relative to the administrative functions that need to be performed, including changing the security characteristics of entities under the control of the TSF.

AGD_ADM.1.7C The administrator guidance shall be consistent with all other documentation supplied for evaluation.

AGD_ADM.1.8C The administrator guidance shall describe all security requirements for the IT environment that are relevant to the administrator.

Evaluator action elements:

AGD_ADM.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.4.4.2 User Guidance (AGD_USR.1)

Dependencies:

ADV_FSP.1 Informal functional specification

Developer action elements:

AGD_USR.1.1D The developer shall provide user guidance.

Content and presentation of evidence elements:

AGD_USR.1.1C The user guidance shall describe the functions and interfaces available to the non-administrative users of the TOE.

AGD_USR.1.2C The user guidance shall describe the use of user-accessible security functions provided by the TOE.

AGD_USR.1.3C The user guidance shall contain warnings about user-accessible functions and privileges that should be controlled in a secure processing environment.

AGD_USR.1.4C The user guidance shall clearly present all user responsibilities necessary for secure operation of the TOE, including those related to assumptions regarding user behavior found in the statement of TOE security environment.

AGD_USR.1.5C The user guidance shall be consistent with all other documentation supplied for evaluation.

AGD_USR.1.6C The user guidance shall describe all security requirements for the IT environment that are relevant to the user.

Evaluator action elements:

AGD_USR.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

## 5.4.5 Life cycle support (ALC)

### 5.4.5.1 Flaw Reporting Procedures (ALC_FLR.2)

Dependencies: No dependencies.

Developer action elements:

ALC_FLR.2.1D The developer shall document the flaw remediation procedures.

ALC_FLR.2.2D The developer shall establish a procedure for accepting and acting upon user reports of security flaws and requests for corrections to those flaws.

Content and presentation of evidence elements:

ALC_FLR.2.1C The flaw remediation procedures documentation shall describe the procedures used to track all reported security flaws in each release of the TOE.

ALC_FLR.2.2C The flaw remediation procedures shall require that a description of the nature and effect of each security flaw be provided, as well as the status of finding a correction to that flaw.

ALC_FLR.2.3C The flaw remediation procedures shall require that corrective actions be identified for each of the security flaws.

ALC_FLR.2.4C The flaw remediation procedures documentation shall describe the methods used to provide flaw information, corrections and guidance on corrective actions to TOE users.

ALC_FLR.2.5C The procedures for processing reported security flaws shall ensure that any reported flaws are corrected and the correction issued to TOE users.

ALC_FLR.2.6C The procedures for processing reported security flaws shall provide safeguards that any corrections to these security flaws do not introduce any new flaws.

Evaluator action elements:

ALC_FLR.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

## 5.4.6 Tests (ATE)

### 5.4.6.1 Evidence of Coverage (ATE_COV.1)

Dependencies:

ADV_FSP.1 Informal functional specification

ATE_FUN.1 Functional testing

Developer action elements:

ATE_COV.1.1D The developer shall provide evidence of the test coverage.

Content and presentation of evidence elements:

ATE_COV.1.1C The evidence of the test coverage shall show the correspondence between the tests identified in the test documentation and the TSF as described in the functional specification.

Evaluator action elements:

ATE_COV.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

### 5.4.6.2 Functional Testing (ATE_FUN.1)

Dependencies: No dependencies.

Developer action elements:

ATE_FUN.1.1D The developer shall test the TSF and document the results.

ATE_FUN.1.2D The developer shall provide test documentation.

Content and presentation of evidence elements:

ATE_FUN.1.1C The test documentation shall consist of test plans, test procedure descriptions, expected test results and actual test results.

ATE_FUN.1.2C The test plans shall identify the security functions to be tested and describe the goal of the tests to be performed.

ATE_FUN.1.3C The test procedure descriptions shall identify the tests to be performed and describe the scenarios for testing each security function. These scenarios shall include any ordering dependencies on the results of other tests.

ATE_FUN.1.4C The expected test results shall show the anticipated outputs from a successful execution of the tests.

ATE_FUN.1.5C The test results from the developer execution of the tests shall demonstrate that each tested security function behaved as specified.

Evaluator action elements:

ATE_FUN.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

#### 5.4.6.3 ATE_IND.2 Independent Testing - Sample

Dependencies:

ADV_FSP.1 Informal functional specification

AGD_ADM.1 Administrator guidance

AGD_USR.1 User guidance

ATE_FUN.1 Functional testing

Developer action elements:

ATE_IND.2.1D The developer shall provide the TOE for testing.

Content and presentation of evidence elements:

ATE_IND.2.1C The TOE shall be suitable for testing.

ATE_IND.2.2C The developer shall provide an equivalent set of resources to those that were used in the developer's functional testing of the TSF.

Evaluator action elements:

ATE_IND.2.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

ATE_IND.2.2E The evaluator shall test a subset of the TSF as appropriate to confirm that the TOE operates as specified.

ATE_IND.2.3E The evaluator shall execute a sample of tests in the test documentation to verify the developer test results.

## 5.4.7 Vulnerability assessment (AVA)

### 5.4.7.1 Examination Of Guidance (AVA_MSU.1)

Dependencies:

ADO_IGS.1 Installation, generation, and start-up procedures

ADV_FSP.1 Informal functional specification

AGD_ADM.1 Administrator guidance

AGD_USR.1 User guidance

Developer action elements:

AVA_MSU.1.1D The developer shall provide guidance documentation.

Content and presentation of evidence elements:

AVA_MSU.1.1C The guidance documentation shall identify all possible modes of operation of the TOE (including operation following failure or operational error), their consequences and implications for maintaining secure operation.

AVA_MSU.1.2C The guidance documentation shall be complete, clear, consistent and reasonable.

AVA_MSU.1.3C The guidance documentation shall list all assumptions about the intended environment.

AVA_MSU.1.4C The guidance documentation shall list all requirements for external security measures (including external procedural, physical and personnel controls).

Evaluator action elements:

AVA_MSU.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_MSU.1.2E The evaluator shall repeat all configuration and installation procedures to confirm that the TOE can be configured and used securely using only the supplied guidance documentation.

AVA_MSU.1.3E The evaluator shall determine that the use of the guidance documentation allows all insecure states to be detected.

#### 5.4.7.2 AVA_SOF.1 Strength Of TOE Security Function Evaluation

Dependencies:

ADV_FSP.1 Informal functional specification

ADV_HLD.1 Descriptive high-level design

Developer action elements:

AVA_SOF.1.1D The developer shall perform a strength of TOE security function analysis for each mechanism identified in the ST as having a strength of TOE security function claim.

Content and presentation of evidence elements:

AVA_SOF.1.1C For each mechanism with a strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the minimum strength level defined in the PP/ST.

AVA_SOF.1.2C For each mechanism with a specific strength of TOE security function claim the strength of TOE security function analysis shall show that it meets or exceeds the specific strength of function metric defined in the PP/ST.

Evaluator action elements:

AVA_SOF.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_SOF.1.2E The evaluator shall confirm that the strength claims are correct.

#### 5.4.7.3 AVA_VLA.1 Developer Vulnerability Analysis

Dependencies:

ADV_FSP.1 Informal functional specification

ADV_HLD.1 Descriptive high-level design

AGD_ADM.1 Administrator guidance

AGD_USR.1 User guidance

Developer action elements:

AVA_VLA.1.1D The developer shall perform a vulnerability analysis.

AVA_VLA.1.2D The developer shall provide vulnerability analysis documentation.

Content and presentation of evidence elements:

AVA_VLA.1.1C The vulnerability analysis documentation shall describe the analysis of the TOE deliverables performed to search for obvious ways in which a user can violate the TSP.

Evaluator action elements:

AVA_VLA.1.1E The evaluator shall confirm that the information provided meets all requirements for content and presentation of evidence.

AVA_VLA.1.2E The evaluator shall conduct penetration testing, building on the developer vulnerability analysis, to ensure obvious vulnerabilities have been addressed.

# End Notes

**1** A deletion of CC text was performed in FDP_ACF.1.2. Rationale: The word "controlled" was deleted from subjects because all subjects are controlled by the TSF.

FDP_ACF.1.2 **Refinement**: The TSF shall enforce the following rules to determine if an operation among ~~controlled~~ subjects and **DBMS-**controlled objects is allowed…

**2** A deletion of CC text was performed in FAU_SAR.1.2. Rationale: the word "user" was replaced with "authorized administrator" because users are not permitted to view audit records, only authorized administrators given explicit read access can view them.

FAU_SAR.1.2 **Refinement**: The TSF shall provide the audit records in a manner suitable for the ~~user~~ **authorized administrator** to interpret the information.

**3** A deletion of CC text was performed in FAU_STG.1.1. Rationale: The word "protect" was replaced with "restrict the deletion of" because it is not the intention to allow deletion of audit records, except for clearing the entire audit log. Also, the words "from unauthorized deletion" were replaced with "in the audit trail to the authorized administrator" because it is the intention that only the authorized administrator is authorized to clear the audit log when it is full.

FAU_STG.1.1 **Refinement**: The TSF shall ~~protect~~ **restrict the deletion of** the stored audit records ~~from unauthorized deletion~~ **in the audit trail to the authorized administrator**.

**4** A deletion of CC text was performed in FAU_STG_EXP.2. Rationale: The words "authorised user with special rights" were replaced with "authorized administrator" to be more specific than the general Common Criteria language.

FAU_STG_EXP.2.1 **Refinement:** The TSF shall provide the **authorized admin**istrator the capability to select one or more of the following actions, *[selection: 'ignore auditable events', 'prevent auditable events, except those taken by the **authorized administrator'**, 'overwrite the oldest stored audit records'*], and *[assignment: other actions to be taken in case of audit storage failure]* to be taken if the audit trail is full.

**5** A deletion of CC text was performed in FPT_RVM.1.1. Rationale: The word "TSF" was replaced by "host OS security functions" to be more specific about the intended support of the IT environment. Also, the word "TSP" was replaced with "host OS security policy" to be more specific about the intended support of the IT environment. The word "TSC" was replaced with "scope of control of the host OS" to be more specific about the intended support of the IT environment.

FPT_RVM.1.1 **Refinement:** The ~~TSF~~ **host OS security functions** shall ensure that ~~TSP~~ **host OS security policy** enforcement functions are invoked and succeed before each function within the ~~TSC~~ **scope of control of the host OS** is allowed to proceed.

**6** A deletion of CC text was performed in FPT_SEP.1.1. Rationale: The word "TSF" was replaced with "security functions of the host OS" to be more specific about the intended support of the IT environment.

FPT_SEP.1.1 **Refinement:** The ~~TSF~~ **security functions of the host OS** shall maintain a security domain for its own execution that protects it from interference and tampering by untrusted subjects.

**7** A deletion of CC text was performed in FPT_SEP.1.2. Rationale: the word "TSF" was replaced with "security functions of the host OS" and the word "TSC" was replaced with "scope of control of the host OS" to be more specific about the intended support of the IT environment.

FPT_SEP.1.2 **Refinement:** The ~~TSF~~ **security functions of the host OS** shall enforce separation between the security domains of subjects in the ~~TSC~~ **scope of control of the host OS**.

**8** A deletion of CC text was performed in FPT_STM.1.1. Rationale: The word "TSF" was replaced with "security functions of the host OS" to be more specific about the intended support of the IT environment.

FPT_STM.1.1 **Refinement:** The ~~TSF~~ **security functions of the host OS** shall be able to provide reliable time stamps for its own use **and for the TOE**.

# 6 Rationale

111 This section provides the rationale for the selection, creation, and use of security objectives and requirements.

## 6.1 Security Objectives derived from Threats

112 Each of the identified threats to security is addressed by one or more security objectives. The table below summarizes this mapping; this is then followed by explanatory text of how the mapping was derived for each threat.

**Table 12 - Mapping of Security Objectives to Threats**

| Threat | Security Objective(s) Addressing Threat |
|---|---|
| T. ADMIN_ERROR | O.ADMIN_GUIDANCE<br>O.INSTALL<br>O.MANAGE |
| T.AUDIT_COMPROMISE | OE.PHYSICAL<br>OE.SELF_PROTECTION<br>O.AUDIT_GENERATION<br>O.AUDIT_PROTECTION |
| T.IMPROPER_INSTALLATION | OE.CONFIG<br>O.ADMIN_GUIDANCE<br>O.INSTALL |
| T.INSECURE_START | O.ADMIN_GUIDANCE<br>O.MANAGE |
| T.MASQUERADE | O.USER_AUTHENTICATION<br>O.USER_IDENTIFICATION |
| T.POOR_DESIGN | O.SOUND_DESIGN<br>O.TESTING |
| T.POOR_IMPLEMENTATION | O.SOUND_IMPLEMENTATION<br>O.TESTING |

| Threat | Security Objective(s) Addressing Threat |
|---|---|
| T.POOR_TEST | O.TESTING |
| T.RESIDUAL_DATA | O.RESIDUAL_INFORMATION |
| T.SYSACC | OE.PHYSICAL<br>O.ACCESS<br>O.ADMIN_GUIDANCE<br>O.MANAGE<br>O.USER_AUTHENTICATION<br>O.USER_IDENTIFICATION |
| T.TSF_COMPROMISE | OE.PHYSICAL<br>OE_TOE_PROTECTION |
| T.UNATTENDED_SESSION | O.PROTECT<br>O.ACCESS<br>O.TRAINED_USER<br>O.USER_AUTHENTICATION |
| T.UNAUTH_ACCESS | OE.PHYSICAL<br>OE.SELF_PROTECTION<br>O.ACCESS<br>O.DISCRETIONARY_ACCESS<br>O.INTERNAL_TOE_DOMAINS<br>O.PROTECT |
| T.UNDETECTED_ACTIONS | OE.PHYSICAL<br>OE.TIME<br>O.AUDIT_GENERATION<br>O.AUDIT_PROTECTION |
| T.UNIDENTIFIED_ACTIONS | O.ADMIN_GUIDANCE<br>O.AUDIT_REVIEW<br>O.MANAGE |

113 **T. ADMIN_ERROR** – *A authorized administrator may incorrectly install or configure the TOE resulting in ineffective security mechanisms.*

114 Improper administration could result if the authorized administrator is unknowledgeable or if the TOE does not provide the proper administration tools. There is always the possibility that the administrator will make an honest mistake. This threat should be mitigated as long as the TOE provides the necessary administrator support (O.MANAGE) and the authorized administrator is provided with knowledge necessary to carry out administrative duties (O.ADMIN_GUIDANCE). The authorized administrator is provided with necessary installation instructions from the developer that details how to securely install the TOE (O.INSTALL).

115 **T.AUDIT_COMPROMISE** - *A malicious process or user may cause audit data to be inappropriately accessed (viewed, modified or deleted), or prevent future records from being recorded, thus masking an attacker's actions.*

116 The TOE will generate an audit log (O.AUDIT_GENERATION). The environment must address the possible compromise of audit data due to physical means (OE.PHYSICAL). The IT environment must also protect itself and its assets (OE.SELF_PROTECTION). The TOE must also provide protection for its audit data (O.AUDIT_PROTECTION).

117 **T.IMPROPER_INSTALLATION** – *The TOE may be delivered, installed, or initially configured in a manner that undermines TOE security.*

118 This threat is addressed by ensuring the appropriate installation guidance is provided to properly and securely install the TOE (O.INSTALL), and that authorized administrators performing the installation have adequate knowledge on how to install the TOE properly and securely (O.ADMIN_GUIDANCE). Care must be taken when installing the TOE to ensure the configuration settings are as specified in the installation guidance for proper, secure installation (OE.CONFIG).

119 **T.INSECURE_START** - *Reboot may result in insecure state of the TOE.*

120 This threat is addressed by ensuring that the authorized administrators have the knowledge necessary to start the system in a secure state.

121 **T.MASQUERADE** - *An unauthorized user, process, or external IT entity may masquerade as an authorized entity to gain access to data or TOE resources.*

122 Addressing the threat of a process or user masquerading as a different process or user produces an objective of uniquely identifying each user (O.USER_IDENTIFICATION). Unique user identification must be supported by the objective of requiring all users of the TOE to prove their claimed identity (O.USER_AUTHENTICATION).

123 **T.POOR_DESIGN** - *Unintentional or intentional errors in requirement specification, design or development of the TOE may occur.*

124 Faults in the TOE's design can be reduced by eliminating errors in the design through the use of sound design principles and documentation of the TOE design (O.SOUND_DESIGN). Design flaws can be mitigated through discovery resulting from testing the implementation (O.TESTING).

125 **T.POOR_IMPLEMENTATION** - *Unintentional or intentional errors in implementing the design of the TOE may occur.*

126 Testing the security functions of the TOE (O.TESTING) can discover implementation errors and show whether the implementation is a faithful instantiation of its design (O.SOUND_ IMPLEMENTATION).

127 **T.POOR_TEST** - *Incorrect system behavior may result from inability to demonstrate that all functions and interactions within the system operation are correct.*

128 This threat deals with the sufficiency of security tests to show that the TOE security functions behave correctly. Addressing this threat requires the developer to demonstrate that adequate testing methods are used that exercise security features. (O.TESTING).

129 **T.RESIDUAL_DATA** - *A user or process may gain unauthorized access to data through reallocation of TOE resources from one user or process to another.*

130 When data is deleted from memory or storage (e.g., disk drive) it is often left intact and not truly erased. Subsequent users who have that same memory space allocated to their processes might be able to observe other users' data that is residual in that memory/storage. Addressing this threat yields the objective that prohibits users from accessing data that had been stored in system resources previously allocated to other users (O.RESIDUAL_INFORMATION).

131 **T.SYSACC** - *A malicious process or user may gain unauthorized access to the authorized administrator account, or that of other trusted personnel.*

132 The threat of the wrong individual gaining unauthorized access to the authorized administrator's account (O.ACCESS) may be addressed by physical means (OE.PHYSICAL), such as in cases where the authorized administrator console is behind a locked door. For other cases, the threat may be mitigated by requiring the authorized administrator to be uniquely identified (O.USER_IDENTIFICATION) and authenticated (O.USER_AUTHENTICATION). Authorized administrators will have to know (O.ADMIN_GUIDANCE) to check this information at each login. The authorized administrator must also be aware that he/she must protect the authentication information that allows access to the authorized administrator account (O.ADMIN_GUIDANCE). The TOE will provide mechanisms for the authorized administrator to set the security attributes for users so they are not allowed admin access (O.MANAGE).

133 **T.TSF_COMPROMISE** - *A user or process may cause, through an unsophisticated attack, TSF data, or executable code to be inappropriately accessed (viewed, modified, or deleted).*

134 The IT environment will protect the TSF data and executable code from a compromise through physical means (OE.PHYSICAL). The TSF data and executable code is protected under the environmental objective for TOE protection (OE.TOE_PROTECTION)(

135 **T.UNATTENDED_SESSION -** *A user may gain unauthorized access to an unattended session.*

136 Unattended sessions must be protected (O.PROTECT) from unauthorized access (O.ACCESS). The TOE must meet objectives for detecting when sessions are unattended and preventing access to those sessions, unless the user re-authenticates. This might be accomplished by simply alerting users that they must not leave sessions unattended (O.TRAINED_USERS) or by requiring users to re-authenticate themselves (O.USER_AUTHENTICATION) after returning to the unattended session.

137 **T.UNAUTH_ACCESS** - *A user may gain unauthorized access (view, modify, delete) to user data.*

138 The threat of unauthorized physical access is addressed by the environment (OE.PHYSICAL). Addressing the threat of other unauthorized access results in the objective of protecting the user data (O.PROTECT). The TOE must satisfy the objective of ensuring that only authorized users may gain access to the TOE and the resources it protects, and that users are not allowed to access protected data for which they are not authorized (O.ACCESS). Access to TSF data is controlled by a discretionary policy (O.DISCRETIONARY_ACCESS). The TOE maintains internal domains to keep data and processes of concurrent users separate, so users cannot observe or interfere with other users' data or queries (O.INTERNAL_TOE_DOMAINS).

139 **T.UNDETECTED_ACTIONS** - *Failure of the IT operating system to detect and record unauthorized actions may occur.*

140 The threat of undetected physical manipulation of the TOE is addressed by the physical protection in the environment (OE.PHYSICAL). Other actions are detected and a record is made (O.AUDIT_GENERATION) including timestamps (OE.TIME). However, it is important to understand that since this protection profile is at the Basic Robustness level, only the minimum level of audit generation is required, which is commensurate with Basic Robustness. To prevent removing evidence of unauthorized actions, the audit records need to be protected from unauthorized modification (O.AUDIT_PROTECTION).

141 **T.UNIDENTIFIED_ACTIONS** - *Failure of the authorized administrator to identify and act upon unauthorized actions may occur.*

142 The threat of a authorized administrator failing to know about malicious audit events produces the objectives of the authorized administrator having the facilities (O.MANAGE) to review audit records (O.AUDIT_REVIEW) and knowing how to do so (O.ADMIN_GUIDANCE).

## 6.2 Objectives derived from Security Policies

143 Each of the identified security policies implies a set of security objectives to be met. The table below summarizes this mapping; this is then followed by explanatory text of how this mapping was derived for each policy.

**Table 13 - Mapping of Security Objectives to Security Policies**

| Policies | Objectives Enforcing Policies |
|---|---|
| P.ACCOUNTABILITY | OE.TIME<br>O.AUDIT_GENERATION<br>O.AUDIT_REVIEW<br>O.USER_IDENTIFICATION |
| P.AUTHORIZATION | O.ACCESS<br>O.PROTECT<br>O.USER_IDENTIFICATION |
| P.AUTHORIZED_USERS | O.ACCESS |

| Policies | Objectives Enforcing Policies |
|---|---|
| P.I_AND_A | O.USER_AUTHENTICATION<br>O.USER_IDENTIFICATION |
| P.INDEPENDENT_TESTING | O.TESTING |
| P.NEED_TO_KNOW | O.ACCESS<br>O.DISCRETIONARY_ACCESS<br>O.PROTECT<br>O.USER_IDENTIFICATION |
| | |
| P.REMOTE_ADMIN_ACCESS | O.ADMIN_GUIDANCE<br>O.MANAGE<br>O.USER_AUTHENTICATION<br>O.USER_IDENTIFICATION<br>OE.SECURE_COMMS |
| P.ROLES | O.ADMIN_ROLE |

144 **P.ACCOUNTABILITY** - *The users of the TOE shall be held accountable for their actions within the TOE.*

145 Enforcement of this policy requires all users to be uniquely identified (O.USER_IDENTIFICATION), their actions be recorded (O.AUDIT_GENERATION) with accurate timestamps (OE.TIME), and the resulting records of their actions be available for review by the authorized administrator (O.AUDIT_REVIEW).

146 **P.AUTHORIZATION** - *The TOE shall limit the extent of each user's abilities in accordance with the TSP.*

147 This policy requires that users in each of the different roles have a set of abilities defined according to the role, which restricts access to user data by users (O.PROTECT, O.ACCESS). Enforcing this policy requires the user to be uniquely identified (O.USER_IDENTIFICATION).

148 **P.AUTHORIZED_USERS** – *Access controls will ensure that only those users who have been authorized to access the protected information within the TOE will be able to do so.*

149 The TOE will provide mechanisms to allow only authorized users to access the TOE, mainly Discretionary Access controls (O.ACCESS).

150 **P.I_AND_A** - *All users must be identified and authenticated prior to accessing any controlled resources with the exception of public objects.*

151 This policy requires users to claim (O.USER_IDENTIFICATION) and verify (O.USER_AUTHENTICATION) their unique identity prior to accessing the TOE.

152 **P.INDEPENDENT_TESTING** - *The TOE must undergo independent testing as part of an independent vulnerability analysis.*

153 This policy requires that independent testing (O.TESTING) be performed.

154 **P.NEED_TO_KNOW** - *The TOE must limit the access to information in protected resources to those authorized users who have a need to know that information.*

155 Enforcement of this policy requires the protection of resources (O.PROTECT) according to the rules of the discretionary access control policy (O.DISCRETIONARY_ACCESS), which controls access based upon the unique identity of users (O.USER_IDENTIFICATION). The authorized administrator will be able to change a user's security attributes when that user no longer needs to access certain information. (O.ACCESS).

156 **P.REMOTE_ADMIN_ACCESS** – *Authorized administrators shall be able to remotely manage the TOE.*

157 For administrators to manage the system (O.MANAGE) remotely there needs to be a protected communications path provided by the environment (OE.SECURE_COMMS). Use of this path is restricted to authenticated (O.USER_AUTHENTICATION) authorized administrators (O.USER_IDENTIFICATION), as described by the administrator guidance (O.ADMIN_GUIDANCE).

158 **P.ROLES -** *The TOE shall provide a authorized administrator role for secure administration of the TOE. This role shall be separate and distinct from other authorized users.*

159 The TOE has the objective of providing an authorized administrator role for secure administration. The TOE may provide other roles as well, but only the role of authorized administrator is required. (O.ADMIN_ROLE)

## 6.3 Objectives derived from Assumptions

160 Each of the identified security assumptions implies a set of security objectives to be met. The table below summarizes this mapping; this is then followed by explanatory text of how this mapping was derived for each assumption.

**Table 14 - Mapping of Security Objectives to Assumptions**

| Assumptions | Objectives Enforcing Assumptions |
|---|---|
| A.NO_EVIL | OE.NO_EVIL<br>OE.CONFIG |
| A.NO_GENERAL_PURPOSE | OE.NO_GENERAL_PURPOSE |
| A.PHYSICAL | OE.PHYSICAL |
| A.ROBUST_ENVIORNMENT | OE.ROBUST_ENVIORNMENT<br>OE.TRUST_IT |
| A.SECURE_COMMS | OE.SECURE_COMMS |

161 **A.NO_EVIL** – *Authorized administrators are non-hostile, appropriately trained, and follow all administrator guidance.*

162 All authorized administrators are trustworthy individuals, having background investigations commensurate with the level of data being protected, have undergone appropriate admin training, and follow all admin guidance (OE.NO_EVIL). Authorized administrators are trusted to properly configure the TOE so it enforces its security policies (OE.CONFIG).

163 **A. NO_GENERAL_ PURPOSE** - *There are no general-purpose computing or storage repository capabilities (e.g., compilers or user applications) available on DBMS servers, other than those services necessary for the operation, administration and support of the DBMS.*

164 The DBMS server must not include any general-purpose commuting or storage capabilities (OE.NO_GENERAL_PURPOSE). This will protect the TSF data from malicious processes.

165 A.PHYSICAL - Physical security, commensurate with the value of the TOE and the data it contains, is assumed to be provided by the IT environment.

166 The TOE, the TSF data, and protected user data is assumed to be protected from physical attack (e.g., theft, modification, destruction, or eavesdropping). Physical attack could include unauthorized intruders into the TOE environment, but it does not include physical destructive actions that might be taken by an individual that is authorized to access the TOE environment (OE.PHYSICAL).

167 **A.ROBUST_ENVIORNMENT** - *It is assumed that the IT environment is at least as robust as the TOE.*

168 The TOE shall only be installed in an IT environment that is at least as robust as the TOE. The TOE is basic robustness, therefore, all elements in the environment the TOE depends on for enforcement of its security objectives are also assumed to be basic robustness. These elements could include the operating system, encryption devices, and/or boundary protection devices (OE.ROBUST_ENVIORNMENT).

169 The IT entities in the environment are correctly installed, configured, managed and maintained (OE.TRUST_IT)

170 **A.SECURE_COMMS** - *It is assumed that the IT environment will have a secure line of communications between the remote user and the TOE.*

171 The environment must provide a secure line of communication for transfer of TSF data (OE.SECURE_COMMS). This is necessary because the TOE may be distributed geographically with users and authorized administrators in different locations. It may also be the case that the TOE is a distributed architecture, with database servers in different geographic locations.

172 The objective OE.SECURE_COMMS does not necessarily mandate that the communications between the remote administrator and the TOE be encrypted. Remote administration implies administration from any location other than the TOE console. In many implementations, remote administration will be done from another workstation on the same LAN as the TOE, but within a protected enclave. In this case, there is no need for cryptographic protection of the communications between the authorized administrator and the TOE.

## 6.4 Requirements Rationale

173 Each of the security objectives identified in sections 6.1 and 6.2 are met by a set of security requirements. The table below summarizes this mapping; this is then followed by explanatory text of how the mapping was derived.

**Table 15 - Mapping of Security Requirements to Objectives**

| Objective | Requirement(s) Addressing the Objective |
|---|---|
| O.ACCESS | FDP_ACC.1, FDP_ACF.1, FMT_REV.1(1), FTA_MCS.1, FTA_TSE.1 |
| O.ADMIN_GUIDANCE | ADO_DEL.1, ADO_IGS.1, AGD_ADM.1, AVA_MSU.1 |
| O.ADMIN_ROLE | FMT_SMR.1 |
| O.AUDIT_GENERATION | FAU_GEN_EXP.1, FAU_GEN_EXP.2, FAU_SEL.1, FIA_USB_EXP.1, FMT_MOF.1, FMT_MTD.1(1), FPT_STM.1 |
| O. AUDIT_PROTECTION | FAU_SAR.2, FAU_STG_EXP.1, FAU_STG_EXP.2, FMT_MOF.1, FMT_MTD.1(2) |
| O. AUDIT_REVIEW | FAU_SAR.1, FAU_SAR.3, FPT_STM.1 |
| O.DISCRETIONARY_ACCESS | FDP_ACC.1, FDP_ACF.1, FDP_ITT.1, FIA_USB_EXP.1, FMT_MSA.1, FMT_MSA.3, FPT_RVM.1 |
| O.INSTALL | ADO_DEL.1, ADO_IGS.1 |
| O.MANAGE | FMT_MOF.1, FMT_MSA.1, FMT_MSA.2, FMT_MSA.3, FMT_MTD.1(1), FMT_MTD.1(2), FMT_MTD.1(3) |
| O.INTERNAL_TOE_DOMAINS | FPT_RVM.1, FPT_ITD_EXP.1 |
| O.PROTECT | FDP_ACC.1, FDP_ACF.1, FDP_ITT.1, FDP_RIP.2, FMT_REV.1(2), FPT_RVM.1, FPT_ITD_EXP.1, FPT_ITT.1, FPT_TRC_EXP.1 |
| | |
| O.RESIDUAL_INFORMATION | FDP_RIP.2 |
| O.SOUND_DESIGN | AVA_MSU.1, AVA_SOF.1, AVA_VLA.1, ADV_FSP.1, ADV_HLD.1, ADV_RCR.1, ADV_SPM.1 |

| Objective | Requirement(s) Addressing the Objective |
|---|---|
| O.SOUND_IMPLEMENTATION | ALC_FLR.2, ATE_COV.1, ATE_FUN.1, ATE_IND.2, AVA_MSU.1, AVA_SOF.1, AVA_VLA.1, ADV_FSP.1, ADV_HLD.1, ADV_RCR.1 |
| O.TESTING | ATE_COV.1, ATE_FUN.1, ATE_IND.2 |
| O.TRAINED_USERS | AGD_USR.1 |
| O.USER_AUTHENTICATION | FIA_AFL_EXP.1, FIA_SOS.1, FIA_UAU.1, FMT_MOF.1, FMT_MSA.2, FMT_MTD.1(3) |
| O.USER_IDENTIFICATION | FIA_ATD.1, FIA_UID.1, FIA_USB_EXP.1 |

174 **O.ACCESS** – *The TOE will ensure that users gain only authorized access to it and to its resources that it controls.*

175 The subjects and objects within the TOE are under the enforcement of a discretionary access control policy. This policy might apply to a subset of the objects under control of the TOE. There may be some objects that are publicly accessible, and not under the control of the Discretionary policy. For example, the database system could have an interface to the Internet that lets users view certain public information using their browser, while protected portions of the database are available only to certain users of the database. Consider a database for a financial institution. Public information might include current rates for savings accounts and for various types of loans. Private information might include information associated with each user's account, such as account balances and status of loan applications. (FDP_ACC.1)

176 The subjects and objects under the discretionary access control policy will have certain rules that apply to all accesses between them. The rules will be based on certain attributes of those subjects and objects. The rules and attributes on which the TSF makes access control decisions is left to be defined by the writer of the Security Target, as is the definition of subjects and objects. The reason for this is that the granularity of access control can vary widely for database systems. For systems based on the relational data model, the TSF might control access on very coarse-grained objects, such as tables, on fine-grained objects such as data rows or even elements, or on derived objects such as data views. Rather than dictate the granularity of access control, this protection profile leaves the granularity open to the particular implementation of the TOE. (FDP_ACF.1)

177 Security attributes associated with subjects and objects are the basis for access control. Revocation of these security attributes would modify the access control policy. The authorized administrator should have control over security attributes associated with users

(such as user authentication data), being the only role that can revoke them. (FMT_REV.1(1))

178 The TOE must keep track of what user sessions are currently established and running, associating each established session with a uniquely identified user. The TOE must allow only one session at a time for a user. (FTA_MCS.1)

179 There may be attributes that would deny establishment of a session to prevent unauthorized use of the TOE by a user. There are many examples that could possibly be here. One example is that authorized users might be prohibited from accessing the TOE after hours or on weekends. This would prevent someone from logging in as another user while that user is not present. (FTA_TSE.1)

180 **O.ADMIN_GUIDANCE** - *The TOE will provide authorized administrators with the necessary information for secure management of the TOE.*

181 When the TOE is delivered for installation, the authorized administrator must have confidence that it is the genuine, unaltered TOE procured from the TOE vendor. Procedures for delivery of the TOE will give the authorized administrator confidence in the TOE, its security mechanisms, and authorized administrator documentation that describes how to perform administrative duties securely. (ADO_DEL.1)

182 Installation and start-up procedures give the authorized administrator information necessary for initial generation of the TOE as intended by the developer. (ADO_IGS.1)

183 "Administrator guidance refers to written material that is intended to be used by those persons responsible for configuring, maintaining, and administering the TOE in a correct manner for maximum security. Because the secure operation of the TOE is dependent upon the correct performance of the TSF, persons responsible for performing these functions are trusted by the TSF. Administrator guidance is intended to help authorized administrators understand the security functions provided by the TOE, including both those functions that require the authorized administrator to perform security-critical actions and those functions that provide security-critical information." [Quoted from: CC v2.1, Part III, Section 11.1] Since this is a software-only TOE, there are some requirements that may be allocated to the IT environment. The host operating system will be depended upon for security support and some security mechanisms. The administrator guidance must exist for the IT environment components that the TOE depends on. (AGD_ADM.1)

184 "The objective is to ensure that misleading, unreasonable and conflicting guidance is absent from the guidance documentation, and that secure procedures for all modes of operation have been addressed. Insecure states should be easy to detect." [Quoted from: CC v2.1, Part III, Section 14.2] This is an assurance requirement for the developer to provide guidance documentation, and for the evaluator to examine the guidance for misleading, unreasonable or conflicting guidance that could hamper secure management of the TOE. (AVA_MSU.1)

185 **O.ADMIN_ROLE** - *The TOE will provide authorized administrator roles to isolate administrative actions.*

186 The TOE will establish, at least, a authorized administrator role. The ST writer may choose to specify more roles. The authorized administrator will be given privileges to perform certain tasks that other users will not be able to perform. These privileges include, but are not limited to, access to audit information and security functions. (FMT_SMR.1)

187 **O.AUDIT_GENERATION** - *The TOE will provide the capability to detect and create records of security relevant events associated with users.*

188 This objective is satisfied in part by the requirement that the TOE generate audit records according to the minimum level of auditing, as defined by the Common Criteria. (FAU_GEN_EXP.1)

189 Each audit record written must be descriptive of the event that caused a record to be generated, and must be associated with the unique identity of the user that caused the event. (FAU_GEN_EXP.2)

190 The TOE enables the authorized administrator to pre-select events to include in the audit log. (FAU_SEL.1)

191 All subjects that act on behalf of users must have a binding that associates the subjects with a user. This is necessary to be able to associate audit records with user identities. (FIA_USB_EXP.1)

192 The TOE ensures that the authorized administrator role is the only role authorized to manipulate the behavior of the audit generation mechanism. (FMT_MOF.1)

193 The TOE allows only authorized administrators to perform pre-selection of auditable events. (FMT_MTD.1(1))

194 Reliable time stamps are assumed to be provided by the IT environment. (FPT_STM.1)

195 **O. AUDIT_PROTECTION** - *The TOE will provide the capability to protect audit information.*

196 Users must not be able to read the audit records, unless they have been granted explicit read-access to the audit log. (FAU_SAR.2)

197 The TOE prevents unauthorized deletion or modification of audit records. (FAU_STG_EXP.1)

198 The TOE provides site-configurable options to prevent loss of audit data in the event the audit storage space is exhausted. (FAU_STG_EXP.2)

199 The TOE ensures that the authorized administrator role is the only role authorized to manipulate the behavior of the audit generation mechanism. (FMT_MOF.1)

200 Only the authorized administrator has the ability to query or clear audit records (FMT_MTD.1(2))

201 **O.AUDIT_REVIEW** - *The TOE will provide the capability to selectively view audit information, and alert the authorized administrator of identified potential security violations.*

202 The authorized administrator will be the only user allowed access to the database audit information. This will prevent unauthorized users from modifying the audit information. In order for the authorized administrator to review the audit logs they must be in a suitable form for the authorized administrator to read, which means the authorized administrator should have the appropriate software and decryption keys needed to interpret the data. (FAU_SAR.1)

203 This requirement can be satisfied by the TOE or by the external IT environment. The authorized administrator must be able to perform queries on the audit data based on date, time, type of event, event status (success or failure), or any other criteria chosen by the ST Writer. This will allow the authorized administrator to search for specific events more efficiently. (FAU_SAR.3)

204 Reliable time stamps are assumed to be provided by the IT environment. The host operating system must provide accurate time stamps for its own use as well as for the TOE. These time stamps will be used for documenting auditing events. (FPT_STM.1)

205 **O.DISCRETIONARY_ACCESS** - *The TOE will control accesses to resources based upon the identity of users or groups of users.*

206 The subjects and objects within the TOE are under the enforcement of a discretionary access control policy. This policy might apply to a subset of the objects under control of the TOE. There may be some objects that are publicly accessible, and not under the control of the Discretionary policy. For example, the database system could have an interface to the Internet that lets users view certain public information using their browser, while protected portions of the database are available only to certain users of the database. Consider a database for a financial institution. Public information might include current rates for savings accounts and for various types of loans. Private information might include information associated with each user's account, such as account balances and status of loan applications. (FDP_ACC.1)

207 The subjects and objects under the discretionary access control policy will have certain rules that apply to all accesses between them. The rules will be based on certain attributes of those subjects and objects. The rules and attributes on which the TSF makes access control decisions is left to be defined by the writer of the Security Target, as is the definition of subjects and objects. The reason for this is that the granularity of access control can vary widely for database systems. For systems based on the relational data model, the TSF might control access on very coarse-grained objects, such as tables, on fine-grained objects such as data rows or even elements, or on derived objects such as data views. Rather than dictate the granularity of access control, this protection profile leaves the granularity open to the particular implementation of the TOE. (FDP_ACF.1)

208 The Discretionary Access Control policy prevents disclosure of user data when it is transmitted between physically separate parts of the TOE (e.g., between the database server and the database client). (FDP_ITT.1)

209 All subjects that act on behalf of users must have a binding that associates the subjects with a user uniquely. (FIA_USB_EXP.1)

210 Only authorized administrators may manipulate the security attributes of database users. (FMT_MSA.1, FMT_MSA.3)

211 The Discretionary Access Control policy is not to be bypassed or optional. The discretionary aspect of the policy is that users who control access to objects can set that access to be restrictive or permissive to other users at their discretion. The policy is to be always enforced, never optional. (FPT_RVM.1)

212 **O.INSTALL** - *The TOE will be delivered with the appropriate installation guidance to establish and maintain TOE security.*

213 The developer must provide and adhere to procedures for secure transfer of the TOE from the development site to the customer's site. This will ensure the TOE is delivered with all necessary security components and is not maliciously modified before it has been installed in the environment. (ADO_DEL.1)

214 The developer must provide the customer with all steps necessary for the secure installation and startup of the TOE. This must include the configuration of the TOE and its initial startup in a secure state. (ADO_IGS.1)

215 **O.INTERNAL_TOE_DOMAINS** - *The TSF will maintain internal domains for separation of data and queries belonging to concurrent users.*

216 The mechanisms providing self-protection are always invoked and not able to be bypassed. (FPT_RVM.1)

217 The TSF enforces separation between the security domains within its scope of control (FPT_ITD_EXP.1)

218 **O.MANAGE** - *The TOE will provide all the functions and facilities necessary to support the authorized administrators in their management of the security of the TOE.*

219 Only the authorized administrator will be able to enable or disable functions of the audit log. This will prevent a malicious user from turning off the audit log while he/she performs a malicious act, then turning it back on when he/she is done. (FMT_MOF.1)

220 Only authorized administrators may manipulate the security attributes of database users. (FMT_MSA.1, FMT_MSA.2, FMT_MSA.3)

221 Only authorized administrators are able to manage the inclusion/exclusion of specific events to be audited. (FMT_MTD.1(1))

222 Only authorized administrators are authorized to query or clear the audit log. (FMT_MTD.1(2))

223 Only authorized administrators are authorized to set or reset user authentication data. FMT_MTD.1(3))

224 **O.PROTECT** - *The TOE will provide mechanisms to protect user data and resources.*

225 The Discretionary Access Control policy applies to all operations between subjects and objects controlled by the TOE. (FDP_ACC.1)

226 The subjects and objects under the discretionary access control policy will have certain rules that apply to all accesses between them. The rules will be based on certain attributes of those subjects and objects. The rules and attributes on which the TSF makes access control decisions is left to be defined by the writer of the Security Target, as is the definition of subjects and objects. The reason for this is that the granularity of access control can vary widely for database systems. For systems based on the relational data model, the TSF might control access on very coarse-grained objects, such as tables, on fine-grained objects such as data rows or even elements, or on derived objects such as data views. Rather than dictate the granularity of access control, this protection profile leaves the granularity open to the particular implementation of the TOE. (FDP_ACF.1)

227 The Discretionary Access Control policy prevents disclosure of user data when it is transmitted between physically separate parts of the TOE (e.g., between the database server and the database client). (FDP_ITT.1)

228 When data is deleted from memory or storage (e.g., disk drive) it is often left intact and not truly erased. It is then possible for other users with access to the memory to view previously

protected data. Therefore when a block of memory is allocated it is necessary to ensure all previous data stored in that block has been made unavailable. (FDP_RIP.2)

229 Security attributes associated with subjects and objects are the basis for protection of objects by the Discretionary Access Control policy. The discretionary nature of the policy allows users to modify access control permissions, which are represented by security attributes. Users are allowed to modify the security attributes of subjects and objects as permitted by the Discretionary Access Control policy. (FMT_REV.1(2))

230 Users will not be able to bypass the security policy in order to enter the TOE. This means they must identify and authenticate themselves before accessing the TOE, and that the Discretionary Access Control policy is always enforced (FPT_RVM.1)

231 The TOE enforces security domains within its scope of control. (FPT_ITD_EXP.1)

232 The TOE must protect all TSF data from modification and disclosure when it is transferred between separate parts of the TOE. (FTP_ITT.1)

233 Replicated TSF data that specifies attributes for access control must be consistent across distributed components of the TOE. The requirement is to maintain consistency of replicated TSF data. (FPT_TRC_EXP.1)

234 **O.RESIDUAL_INFORMATION** - *The TOE will ensure that any information contained in a protected resource is not released when the resource is reallocated.*

235 When data is deleted from memory or storage (e.g., disk drive) it is often left intact and not truly erased. It is then possible for other users with access to the memory to view previously protected data. Therefore when a block of memory is allocated it is necessary to ensure all previous data stored in that block has been made unavailable. (FDP_RIP.2)

236 **O.SOUND_DESIGN** - *The design of the TOE will be the result of sound design principles and techniques, which are accurately documented.*

237 The evaluators examine the developers guidance on configuring the TOE securely. The purpose for the examination of the guidance is to ensure that it is not self-contradictory, confusing or unreasonable. (AVA_MSU.1)

238 The developer's analysis of the strength of the functions of the TSF shows that the functions meet or exceed SOF-basic. (AVA_SOF.1)

239 The developer conducts a vulnerability analysis that shows whether any identified vulnerabilities in the TOE provide an obvious way to circumvent the TSF. (AVA_VLA.1)

240 The developer provides an informal functional specification of the TOE that describes the user-visible interface and behavior of the TSF. (ADV_FSP.1)

241 The developer must document the informal high-level design of the TOE, describing the TOE in terms of major structural units and the security functions each unit provides. (ADV_HLD.1)

242 The correspondence between the various levels of abstraction of the TOE representation shows that there is correspondence between the high-level design and the functional specification. (ADV_RCR.1)

243 The developer has an informal model of the Discretionary Access Control policy that shows correspondence between the functional specifications, the informal policy model and the policy implementation. (ADV_SPM.1)

244 **O.SOUND_IMPLEMENTATION** - *The implementation of the TOE will be an accurate instantiation of its design.*

245 The developer provides an informal functional specification of the TOE that describes the user-visible interface and behavior of the TSF. (ADV_FSP.1)

246 The developer must document the informal high-level design of the TOE, describing the TOE in terms of major structural units and the security functions each unit provides. This will assist the developer in finding any flaws in the design before it is implemented. (ADV_HLD.1)

247 The correspondences between the various levels of abstraction of the TOE representation show that there is correspondence between the high-level design and the functional specification. (ADV_RCR.1)

248 The developer has procedures for remediation of flaws discovered in the TOE. The developer has procedures for user reports of newly discovered flaws. The developer, according to these procedures, acts upon flaws discovered by users. If users discover flaws in the instantiation, the user will report the flaw so the developer can correct them in the next implementation of the TOE. (ALC_FLR.2)

249 The coverage of testing is sufficient to show that the TSF is tested and shown to operate as specified in the functional specification. This will help to reduce implementation flaws. (ATE_COV.1)

250 The functional components of the TSF are tested, and shown to operate as specified. However, there is no assurance that the TSF does not perform operations that are not in the specification. That is, the complete behavior of the TSF might not be reflected in the functional specification, and the testing will therefore not test any functionality that is not in the functional specification. (ATE_FUN.1)

251 An independent party other than the developer conducts testing. This overcomes the risk of incorrect assessment of the test outcomes on the part of the developer. This will help to reduce implementation flaws. (ATE_IND.2)

252 The evaluators examine the developer's guidance on configuring the TOE securely. The purpose for the examination of the guidance is to ensure that it is not self-contradictory, confusing or unreasonable. (AVA_MSU.1)

253 The developer must perform a strength of TOE security function analysis on all mechanisms that hold a strength of function claim. These mechanisms are to be defined by the ST writer. The developer must show it meets or exceeds its strength of function level, which in this case is SOF-basic. (AVA_SOF.1)

254 The developer conducts a vulnerability analysis that shows whether any identified vulnerabilities in the TOE provide an obvious way to circumvent the TSF. This analysis will show the developer if there are any vulnerabilities that he/she will have to fix. (AVA_VLA.1)

255 **O.TESTING** - *The TOE will undergo developer and independent testing and include test scenarios and results.*

256 The developer must show evidence of the test coverage. This must correspond with the tests identified in the test documentation. (ATE_COV.1)

257 The developer must test the TSF and document the results. The documentation must include test plans, procedures, expected results, and actual results. The plans must identify the security functions tested. (ATE_FUN.1)

258 The developer must have the TOE tested by an independent party. The evaluator will test a subset of the TSF and confirm it operates as specified by the developer. The evaluator will then provide the appropriate evidence that it was tested. (ATE_IND.2)

259 **O.TRAINED_USER** - *The TOE will provide authorized users with the necessary guidance for secure use of the TOE, to include secure sharing of user data.*

260 The developer of the TOE must provide appropriate user training in order to avoid misuse of the TOE resulting in a leak of protected data. This training must teach the user about the interfaces of the TOE. It must also include instruction on the security functions provided. They will explain the importance of the security functions in protecting user data. It will explain all responsibilities the user has for secure operation. This will also include training for secure operation of the IT environment. The training will be consistent with all other documentation for the TOE. The training does not need to include instruction on administrative functions. (AGD_USR.1)

261 **O.USER_AUTHENTICATION** - *The TOE will verify the claimed identity of the user.*

262 To prevent brute force attacks on authentication data, the administrator must specify an upper bound on the number of unsuccessful authentications that will be allowed. Surpassing that threshold could indicate a brute force user authentication attack, and the TOE needs to take appropriate action. (FIA_AFL_EXP.1)

263 User authentication is meaningful only if there is an extremely low probability of success for random attempts to authenticate as an authorized user. The requirement that the secret authentication data be computationally difficult to guess randomly (FIA_SOS.1)

264 User authentication may be satisfied in the IT environment or by the TOE. This requirement allows the ST writer to specify what TSF mediated actions (if any) may be performed before the user is authenticated. The required method of authentication is not specified, and may be passwords, biometrics, cryptographic, or other methods. Combinations of authentication methods are also possible. Also, users authorized to access the TOE must identify themselves to the TOE. (FIA_UAU.1)

265 Only authorized administrators may access administrative resources. Specifically, only authorized administrators may manipulate the audit policy by enabling or disabling audit events. (FMT_MOF.1)

266 The security attributes cannot be set to insecure values. Specifically, the security attributes for user authentication is the user authentication data. Most common implementation use passwords for user authentication. An example of an insecure value for user passwords would be to initialize all user passwords for new user accounts to the same value (e.g., changeme). This would allow a rogue user to attempt that value on all user accounts, in a search for a user who has not changed his/her initial password. (FMT_MSA.2)

267 The user authentication data is to be set only by an authenticated individual in an authorized role. Since the user authentication requirement may be satisfied either by the TOE or by the external IT environment, the specific role is not articulated in the requirement. Rather, it is left as an assignment to be made by the author of the security target. (FMT_MTD.1(3))

268 **O.USER_IDENTIFICATION** - *The TOE will uniquely identify users.*

269 Each database user will have a list of security attributes associated with them. They will have their unique identifier, any groups they may be a part of, for discretionary access control, any security roles they posses, and any other attributes assigned by the ST writer. (FIA_ATD.1)

270 User identification may be satisfied in the IT environment or by the TOE. This requirement allows the ST writer to specify what TSF mediated actions (if any) may be performed before

the user is identified. Also, users authorized to access the TOE must identify themselves to the TOE. (FIA_UID.1)

271 All subjects that act on behalf of users must have a binding that associates the subjects with a user uniquely. (FIA_USB_EXP.1)

## 6.5 Rationale for Explicit Requirements

272 Explicit components have been included in this protection profile because the Common Criteria requirements were found to be insufficient as stated. Table 16 includes the rationale for using explicit requirements.

**Table 16 - Rationale for Explicit Requirements**

| Explicit Component | Rationale |
|---|---|
| FAU_GEN_EXP.1 | Using NIAP Interpretation FAU_GEN.1-NIAP-0347 |
| FAU_GEN_EXP.2 | Using NIAP Interpretation FAU_GEN.2-NIAP-0410 |
| FAU_STG_EXP.1 | Using NIAP Interpretation FAU_STG.1-NIAP-0423 |
| FAU_STG_EXP.2 | Using NIAP Interpretation FAU_STG.NIAP-0414 |
| FIA_AFL_EXP.1 | Using NIAP Interpretation FIA_AFL.1-NIAP-0425 |
| FIA_USB_EXP.1 | Using NIAP Interpretation FIA_USB.1-NIAP-0415 |

| Explicit Component | Rationale |
|---|---|
| FPT_TRC_EXP.1 | FPT_TRC_EXP.1 has been created to require timely consistency of replicated TSF data. Although there is a Common Criteria Requirement that attempts to address this functionality, it falls short of the needs of the environment in this protection profile.<br><br>Specifically, FPT_TRC.1.1 states that "The TSF shall ensure that TSF data is consistent when replicated between parts of the TOE." In the widely distributed environment of this PP's TOE, this is an infeasible requirement. For TOEs with a very large number of components, 100 percent TSF data consistency is not achievable and is not expected at any specific instant in time.<br><br>Another concern lies in FPT_TRC.1.2 which states that when replicated parts of the TSF are "disconnected", the TSF shall ensure consistency of the TSF replicated data upon "reconnection". Upon first inspection, this seems reasonable, however, when applying this requirement it becomes clear that it dictates specific mechanisms to determine when a component is "disconnected" from the rest of the TSF and when it is "reconnected". This is problematic in this PP's environment in that it is not the intent of the authors to dictate that distributed TSF components keep track of connected/disconnected components.<br><br>In general, to meet the needs of this PP, it is acceptable to simply require a mechanism that provides TSF data consistency in a timely manner after it is determined that it is inconsistent. |
| FPT_ITD_EXP.1 | Subjects under the control of the software-only TOE must also have their security domains isolated from one another. Concurrent users of the database management system must be sure that their data is not observed or modified by other users of the same system. |
| ACM_CAP_EXP.2 | Using NIAP Interpretation I-0412 (i.e., ACM_CAP.2.2D is deleted. |

## 6.6 Rationale for Strength of Function

273 The TOE minimum strength of function is SOF-basic. The evaluated TOE is intended to operate in DoD basic robustness environments processing classified information. Users in a DoD environment will have a clearance to access all data processed by the TOE, but not necessarily the need to know. All users are assumed to be cooperative and non-malicious. In commercial environments, company sensitive information may be processed, with users

being cooperative, and not likely to attempt sophisticated attacks at data for which they are not authorized.

## 6.7 Rationale for Assurance

274 This protection profile is developed at the basic robustness level. The assurance requirements are those recommended in recommendation 7 from the Protection Profile Consistency Guidance for Basic Robustness, dated 24 July 2002.

## 6.8 Rationale for Not Including Interpretations

275 This protection profile is current with all but two NIAP and International Common Criteria interpretations as of January 24, 2003. The authors made a conscious decision to not include two interpretations:

276 Rationale for not including NIAP-0407 Empty Selections or Assignments: The protection profile authors and reviewers believe this interpretation is not necessary to allow the writer of a security target to exercise no options for selections and assignments. The cumbersome and confusing nested construct of "[selection…[assignment…]] is unnecessary. Hence, this document maintains the original constructs of the Common Criteria version 2.1.

277 Rationale for not including International Interpretation RI #65 for the FMT class: The proposed new family of FMT_SMF, which is intended to allow specification of management functions to be provided by the TOE is unnecessary and redundant with other families within the FMT class. The authors opted to not include this new family to eliminate redundancy within the protection profile.

## 6.9 Rationale for not including cryptography requirements

278 The TOE is not necessarily intended for use in an environment where cryptography is necessary. If the TOE is contained within an enclave, and is distributed over multiple hosts including servers and user clients, the protection of the communication facilities within the enclave obviates the need for cryptography within the DBMS. If data communication must occur between hosts in different enclaves, the DBMS may depend on the IT environment (operating system, VPN devices, etc) for encryption of data communications.

# 7 References

[1] Common Criteria Implementation Board, Common Criteria for Information Technology Security Evaluation, CCIB-98-026, Version 2.1, August 1999

[2] Department of Defense Chief Information Officer, Guidance and Policy for Department of Defense Information Assurance Memorandum No. 6-8510 dated 16 June 2000

[3] National Security Agency, Protection Profile For Single-level Operating Systems In Environments Requiring Medium Robustness Version 1.22, 23 May 2001

[4] Department of Defense Standard, Department of Defense Trusted Computer System Evaluation Criteria (Orange Book), December 1985

[5] Trusted Product Evaluation Program (TPEP) Trusted Computer System Evaluation Criteria (TCSEC) Interpretations

[6] National Computer Security Center, Trusted Database Management System Interpretation of the Trusted Computer System Evaluation Criteria, NCSC-TG-021 Version-1, April 1991

[7] Security Agency Information Assurance Solutions Technical Directors, Information Assurance Technical Framework, Release 3.0, National September 2000

[8] Protection Profile for Operating Systems Implementing Commercial Security, Version 1.0, dated 27 December 2001

[9] Protection Profile Review Board, Protection Profile Consistency Guidance for Basic Robustness, Version 1.1, dated 1 September 2002